

EAST Search History

| Ref # | Hits | Search Query | DBs | Default Operator | Plurals | Time Stamp |
|-------|------|---|--|------------------|---------|------------------|
| L1 | 5 | GUI and (web or internet) and insurance.ti,ab. | EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2007/09/04 11:05 |
| L2 | 0 | (legacy or mainframe) and (GUI or interface) and (web or internet) and (remote with access) and insurance | EPO; JPO | OR | OFF | 2007/09/04 11:07 |
| L3 | 0 | (legacy or mainframe) and (GUI or interface) and (web or internet) and (remote with access) and insurance | EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2007/09/04 11:10 |
| L4 | 15 | ("20020111725" "20020111835" "20020188484" "20030093302" "5655085" "5768578" "5870719" "5873066" "5926817" "5956691" "5978799" "5987434" "6078890" "6144944" "6175831").PN. | EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2007/09/04 11:10 |
| L5 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2007/09/04 11:12 |
| L6 | 136 | ("3970992" "4347568" "4359631" "4491725" "4553206" "4567359" "4598367" "4633430" "4642768" "4646250" "4730252").PN. OR ("4831526").URPN. | US-PGPUB; USPAT; USOCR | OR | OFF | 2007/09/04 11:13 |
| L7 | 0 | (legacy or mainframe) and (GUI or interface) and insurance | EPO; JPO | OR | OFF | 2007/09/04 11:17 |
| L8 | 29 | (GUI or interface) and insurance | EPO; JPO | OR | OFF | 2007/09/04 11:17 |
| S3 | 454 | (legacy or mainframe) and (GUI or interface) and (web or internet) and (remote with access) and insurance | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2007/09/04 11:17 |
| S6 | 17 | (legacy or mainframe) and (GUI or interface) and (web or internet) and (remote with access) and insurance.ti,ab. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/05 16:53 |
| S7 | 11 | (legacy or mainframe) and (GUI or interface) and (web or internet) and (remote with access) and insurance and 705/4.ccls. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/06 16:52 |

EAST Search History

| | | | | | | |
|-----|----|--|---|----|-----|------------------|
| S8 | 14 | (legacy or mainframe) and (GUI or interface) and (web or internet) and (remote with access) and insurance and 600/300.ccls. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/06 15:34 |
| S9 | 12 | ((("6144990") or ("6122632") or ("5974430") or ("5793964") or ("5754830") or ("5701451") or ("5191522") or ("4831526") or ("5182705") or ("5262940") or ("5493105") or ("6615258"))).PN. | US-PGPUB; USPAT | OR | OFF | 2005/10/06 16:24 |
| S10 | 0 | S9 and (access with den\$4) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/06 16:27 |
| S11 | 3 | (S7 or S8) and (access with den\$4) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/06 16:28 |
| S13 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2005/10/06 16:29 |
| S14 | 4 | 09/941841.app. 10/918967.app. 09/785462.app. 10/240613.app. | US-PGPUB; USPAT | OR | OFF | 2005/10/06 16:31 |
| S15 | 7 | S14 or S13 | US-PGPUB; USPAT | OR | OFF | 2005/10/06 16:32 |
| S16 | 7 | S15 and access | US-PGPUB; USPAT | OR | OFF | 2005/10/06 16:50 |
| S17 | 1 | S15 and GUI | US-PGPUB; USPAT | OR | OFF | 2005/10/06 16:51 |
| S18 | 79 | GUI and (web or internet) and insurance.ti,ab. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2007/09/04 11:05 |
| S19 | 41 | GUI and (web or internet) and insurance.ti,ab. and 705/4.ccls. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/06 16:53 |

EAST Search History

| | | | | | | |
|-----|----|--|---|----|-----|------------------|
| S20 | 17 | GUI.ti,ab. and insurance.ti,ab. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/06 16:54 |
| S21 | 3 | GUI.ti,ab. and insurance.ti,ab. | US-PGPUB; USPAT | OR | OFF | 2005/10/06 16:55 |
| S22 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2005/10/12 09:56 |
| S23 | 4 | 09/941841.app. 10/918967.app. 09/785462.app. 10/240613.app. | US-PGPUB; USPAT | OR | OFF | 2005/10/12 09:56 |
| S24 | 7 | S23 or S22 | US-PGPUB; USPAT | OR | OFF | 2005/10/12 09:56 |
| S25 | 3 | S24 and mainframe | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/12 13:05 |
| S26 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2005/10/12 13:05 |
| S27 | 4 | 09/941841.app. 10/918967.app. 09/785462.app. 10/240613.app. | US-PGPUB; USPAT | OR | OFF | 2005/10/12 13:05 |
| S28 | 7 | S27 or S26 | US-PGPUB; USPAT | OR | OFF | 2005/10/12 13:05 |
| S29 | 7 | S28 and (custom\$5 or personal\$5) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/12 14:55 |
| S30 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2005/10/12 14:55 |
| S31 | 4 | 09/941841.app. 10/918967.app. 09/785462.app. 10/240613.app. | US-PGPUB; USPAT | OR | OFF | 2005/10/12 14:55 |
| S32 | 7 | S31 or S30 | US-PGPUB; USPAT | OR | OFF | 2005/10/12 14:55 |
| S33 | 4 | S32 and search | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2005/10/12 14:55 |
| S34 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2005/10/13 11:58 |

EAST Search History

| | | | | | | |
|-----|-----|---|---|----|-----|------------------|
| S35 | 4 | 09/941841.app. 10/918967.app. 09/785462.app. 10/240613.app. | US-PGPUB; USPAT | OR | OFF | 2005/10/13 11:58 |
| S36 | 7 | S35 or S34 | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:41 |
| S37 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:40 |
| S38 | 4 | 09/941841.app. 10/918967.app. 09/785462.app. 10/240613.app. | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:41 |
| S39 | 4 | 09/941841.app. 10/918967.app. 09/785462.app. 10/240613.app. | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:41 |
| S40 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:42 |
| S41 | 7 | S39 or S40 | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:42 |
| S42 | 0 | S41 and (java or wrapper) | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:43 |
| S43 | 1 | S41 and gui | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:48 |
| S44 | 522 | (legacy or mainframe) and (GUI or interface) and (web or internet) and (remote with access) and insurance | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | OFF | 2006/04/26 12:48 |
| S45 | 57 | S44 and wrapper | US-PGPUB; USPAT | OR | OFF | 2006/04/26 12:54 |
| S48 | 4 | 09/941841.app. 10/918967.app. 09/785462.app. 10/240613.app. | US-PGPUB; USPAT | OR | OFF | 2006/04/26 14:44 |
| S49 | 3 | ((("6922720") or ("5191522") or ("4831526"))).PN. | US-PGPUB; USPAT | OR | OFF | 2006/04/26 14:44 |
| S50 | 7 | S48 or S49 | US-PGPUB; USPAT | OR | OFF | 2006/04/26 14:44 |
| S51 | 6 | S50 and (bill\$4 or premium) | US-PGPUB; USPAT | OR | OFF | 2006/04/26 14:45 |

Logon

*** It is now 9/4/2007 9:56:33 AM ***

Welcome to DialogLink - Version 5

Revolutionize the Way You Work!

New on Dialog

Enhanced Derwent World Patents Index Now Available

The enhanced *Derwent World Patents Index*[®] (*DWPI*SM) (Files 350,351,352) is now available on Dialog. The improvements implemented in *DWPI* on Dialog further extend the database's rich content set and enhances overall functionality of the database.

In addition to distilled expert analysis reflected in *DWPI* expanded titles and abstracts, other enhancements include original patent filing details, multiple patent images, easy cut-and-paste patent family data, and much more.

The new templates include new features that will help you manage and distribute your *DWPI* search results in an attractive format.

Learn about all of the new *DWPI* enhancements and report templates at <http://www.dialog.com/dwpi>.

DialogLink 5 Release Notes

New features available in the latest release of DialogLink 5 (November 2005)

- Ability to resize images for easier incorporation into DialogLink Reports
- New settings allow users to be prompted to save Dialog search sessions in the format of their choice (Microsoft Word, RTF, PDF, HTML, or TEXT)
- Ability to set up Dialog Alerts by Chemical Structures and the addition of Index Chemicus as a structure searchable database
- Support for connections to STN Germany and STN Japan services

Show Preferences for details

? Help Off Line

* * *

Connecting to shayglass - Dialog - 290603

Connected to Dialog via SMS003196000

? b 411

Estimated Cost Summary

| Project | Client | Charge Code | Searcher | Job | Service | User |
|---------|--------|-------------|----------|-----|---------|------|
|---------|--------|-------------|----------|-----|---------|------|

| | | | | | | | | | | |
|------------------|-------------------|----------------------|---------------------|------------------|---------------|-------------------|-------------|-------------------|-------------|---------------|
| | | | | | | | | | Code | Number |
| | | | | | | shayglass | | | 51 | 290603 |
| Date | | Time | | SessionID | | Subsession | | Subaccount | | |
| 09/04/2007 | | 08:58:30 | | 2 | | 3 | | | | |
| Data Base | Dial Units | Access Charge | Print Credit | Types | Prints | Report | Rank | Links | CSS | Total |
| 411 | 0.1130 | 0.33 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.33 |
| Sub Totals | 0.1130 | \$0.33 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.33 |
| Session Totals | 0.4750 | \$0.51 | | Telecom | \$0.32 | | | | | \$0.83 |

Begin insure? (s) wrapper (s) (GUI or interface) (s) legacy
>>>W: "INSURE?" is invalid in a filelist.
No valid file banners found
>>>E: You may not BEGIN more than one service at a time

? b 411

Estimated Cost Summary

| Estimated Cost Summary | | | | | | | | | | | |
|------------------------|------------|---------------|--------------|-------------|--------|------------|--------|------------|--------|--------------|-------------|
| Project | | Client | | Charge Code | | Searcher | | Job | | Service Code | User Number |
| | | | | | | shayglass | | | | 51 | 290603 |
| Date | | Time | | SessionID | | Subsession | | Subaccount | | | |
| 09/04/2007 | | 09:01:17 | | 2 | | 5 | | | | | |
| Data Base | Dial Units | Access Charge | Print Credit | Types | Prints | Report | Rank | Links | CSS | Total | |
| 411 | 0.1150 | 0.34 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.34 | |
| Sub Totals | 0.1150 | \$0.34 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.34 | |
| Session Totals | 1.0470 | \$1.98 | | Telecom | \$0.26 | | | | | \$2.23 | |

>>>E: No databases were chosen

? b 411

> Set Files all
> Select legacy (s) wrapper (s) (gui or interface) (s) insure?
>>>W: I/O error in file 220
Processing
4 databases have items, of 564 searched.

Hits File Name

- 1 148 Gale Group Trade & Industry DB 1976-2007/Aug 28
- 15 654 US PAT.FULL. 1976-2007/AUG 30
- 1 761 Datamonitor Market Res. 1992-2007/Aug
- 1 991 NewsRoom 2006 Jan 1-2006/Dec 31

Estimated Cost Summary

| Project | | Client | | Charge Code | | Searcher | | Job | | Service Code | User Number |
|----------------|------------|---------------|--------------|-------------|--------|------------|--------|------------|--------|--------------|-------------|
| | | | | | | shayglass | | | | 51 | 290603 |
| Date | | Time | | SessionID | | Subsession | | Subaccount | | | |
| 09/04/2007 | | 09:09:03 | | 2 | | 7 | | | | | |
| Data Base | Dial Units | Access Charge | Print Credit | Types | Prints | Report | Rank | Links | CSS | Total | |
| 411 | 8.1820 | 24.06 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 24.06 | |
| Sub Totals | 8.1820 | \$24.06 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$0.00 | \$24.06 | |
| Session Totals | 9.5720 | \$26.85 | | Telecom | \$1.82 | | | | | \$28.67 | |

Begin 148,654,761,991

[File 148] **Gale Group Trade & Industry DB 1976-2007/Aug 28**

(c)2007 The Gale Group. All rights reserved.

**File 148: The CURRENT feature is not working in File 148. See HELP NEWS148.*

[File 654] **US PAT.FULL. 1976-2007/AUG 30**

(c) Format only 2007 Dialog. All rights reserved.

**File 654: IPCR/8 classification codes now searchable in 2006 records. For information about IC= index changes, see HELP NEWSIPCR.*

[File 761] **Datamonitor Market Res. 1992-2007/Aug**

(c) 2007 Datamonitor. All rights reserved.

[File 991] **NewsRoom 2006 Jan 1-2006/Dec 31**

(c) 2007 Dialog. All rights reserved.

SELECT legacy (s) wrapper (s) (gui or interface) (s) insure?

379173 LEGACY

32981 WRAPPER

78038 GUI

1428778 INTERFACE

795015 INSURE?

S1 18 SELECT LEGACY (S) WRAPPER (S) (GUI OR INTERFACE) (S) INSURE?

?

? t s1/3,k/all

1/3,K/1 (Item 1 from file: 148) Links

Gale Group Trade & Industry DB

(c)2007 The Gale Group. All rights reserved.

0021073284 **Supplier Number:** 145268281 (USE FORMAT 7 OR 9 FOR FULL TEXT)

From promises to prime time: early adopters got into the Web-services game a while ago, but most insurers took a wait-and-see stance and held off for proof of performance. The time may now be ripe to get on board--but jumping on without a good strategy could make for a rough ride.(TECHNOLOGY)

Meyer, David

Risk & Insurance , 17 , 5 , 74(2)

April 15 , 2006

ISSN: 1050-9232

Language: English

Record Type: Fulltext

Word Count: 1582 **Line Count:** 00134

...recently, the majority of Web services have been implemented in their simplest form with some **insurers** using services as a **wrapper** for their **legacy** system to expose the older technology to the Internet--a technique commonly referred to as Web-enabling. While the resulting user **interface** may be based on Web services, the underlying technology remains the **legacy** architecture with all its maintenance and integration challenges.

Web services have also been commonly used...

1/3,K/2 (Item 1 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

5882020 **IMAGE Available

Derwent Accession: 2005-009938

Utility

System and method for dynamically adjusting to CPU performance changes

Inventor: Cooper, Barnes, Aloha, OR

Assignee: Intel Corporation 02), Santa Clara, CA

Examiner: Wiley, David (Art Unit: 213)

Assistant Examiner: Avellino, Joseph E.

Law Firm: Schwegman, Lundberg, Woessner & Kluth, P.A.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6823516 | A | 20041123 | US 99371751 | 19990810 |

Fulltext Word Count: 18816

Description of the Invention:

...support mechanism for performance state transitions consists of the following two software elements: a user **interface** and a device driver. In one embodiment, the user **interface** is an extension of the operating system's configuration/power management application (e.g., the...

...In one embodiment a WDM device driver receives mode commands from the user **interface**. In one embodiment, mode settings are stored in the system registry. The driver registers for...

...When using SMIs to implement performance state transitions in **legacy** operating systems, it is recommended that they be set up on all pertinent events (e...as they are received. Once a performance state transition application has been granted the transition **interface**, the application periodically queries the SMI handler for system status. These status calls return information...

...one such embodiment, the application reads the user configuration settings from a control panel user **interface**. By using all of this data, the application can then request that the SMI handler...

...the application. The application can then note the new performance state and update the user **interface**, or retry the operation at a later time if it failed...

...to 170 and SMM support for performance state transitions is disabled, as

is the application **interface**.

...

...moves to 174, where SMM support for performance state transitions is disabled and the application **interface** is enabled...

...disable command. If so, control moves to 180, the application is disabled and the SMI **interface** is enabled...

...SMI handler to cease managing performance state transitions, as well as effectively disabling the SMI **interface** to the performance state transition application. If the application is already loaded, the application receives...

...application loads after the OS has enabled native transition support, the enable performance transition application **interface** will fail the request. In either case, the application will unload once the OS has ...

...routine with the segment registers setup appropriately. In one embodiment this is handled by a **wrapper** routine that performs the segment loading...

...tasks for managing a performance-state-enabled system. The first is that it provides the **interface** to the Performance State Transition Applet for reading status, checking and setting capabilities, and performing...

...The following components are covered in this implementation: the Performance State Transition Applet SMM **Interface** and the State Transition Event Handlers. The Performance State Transition Applet SMM **Interface** routine processes software SMIs from the Performance State Transition Applet, or in response to transition...

...In one embodiment, the Performance State Transition Applet SMM **Interface** is accessed as a hook to the processing of generic software SMIs. In one such...

...values stored in the CPU's general-purpose registers. The next section describes this command **interface** in detail...

...mode, as the OS or applet is in control. When neither the OS nor applet **interface** is in control, the SMM handler can perform performance state transitions in response to AC...

...WINDOWS NT 4.0, WINDOWS 98 and WINDOWS 2000 operating system via an SMM-based **interface**. An applet is provided that communicates to the platform system management interrupt (SMI) handler to...

...one such embodiment, the SMI handler performs all performance state transitions so that it can **insure** that OS registers are properly saved/restored in order to prevent collision with accesses by...

...are applied to operating systems other than WINDOWS 98 and WINDOWS 2000.

This is because **legacy** non-ACPI compliant operating systems used CPU speed dependent calibration loops for driver timing routines...

...3) SMI+Applet (**legacy** non-ACPI compliant operating systems). A one-time transition from high to low frequency is...

...The second SMI command port value is used for the performance state transition applet **interface**. In this case, in order to claim only one additional SMI command port value and...

...To keep the **interface** isolated from collisions from ACPI OS commands, in one embodiment the **interface** uses only one register to initiate the SMIs. In addition, returned data is reported back...

...In one embodiment, the following commands can be used to communicate with the SMM **interface**: Performance State Disable, Performance State Control, Get Performance State Status, Set Performance State, and Set...

...this command has two subfunctions that allow the applet to enable and disable the SMM **interface**. However, the Get Performance State Status function is always enabled...

...which setup mode the user has selected, AC status, and the revision of the SMM **Interface** it is compliant with. (NOTE: This function must maintain memory copies of all variables to...

...user BIOS setup options to reflect changes made by the user in the applet user **interface**. The setup options include disabling performance state transitions, operating in one particular state all of

1/3,K/3 (Item 2 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

5556067 **IMAGE Available

Derwent Accession: 2001-191263

Utility

E/ **Braille computer monitor**

Inventor: Becker, John V., late of New York, NY, deceased

Becker, David A., 87 Remsen St., Brooklyn, NY, 11201, legal representative

Hinton, Daniel E., 815 Galway Garth, Arnold, MD, 21012-1342

Anderson, Jr., Hugh G., 1755 Manchester Rd., Westminster, MD, 21157

Assignee: Unassigned

Unassigned Or Assigned To Individual (Code: 68000)

Examiner: Liang, Regina (Art Unit: 264)

Law Firm: Katten Muchin Zavis Rosenman

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6700553 | A | 20040302 | US 2002131542 | 20020424 |
| Division | US 6417821 | A | | US 2000605514 | 20000628 |

Fulltext Word Count: 5657

Description of the Invention:

...belt power transmission system 800, and a support structure 900. Not shown are a computer **interface**, an outer casing, a computer and an electronic control package of generally known arrangement...

...100 have a spring that returns them to the raised position. The twice tapping requirement **insures** intentional versus accidental communicative desires and is similar to double clicking with a mouse...

...deflections imposed by dot pin drag forces. Part of the difficulty was from working with **legacy** dot pin dimensions. Therefore, in configuring an erasing mechanisms 400 especially for multi-line TCMs...

...The user **interface** for the erasing mechanism 400 is a hand operated electronic switch that is surface mounted...

...A standard computer **interface** is connector-mounted on the traveling printhead 300 and connects the TCM to a standard...

...wired to the connector. The outer casing of the TCM is a conformal shell or **wrapper** that provides a smooth **interface** for the user while preventing the user from exposure to the inner workings of the...

1/3,K/4 (Item 3 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

5364989 **IMAGE Available

Derwent Accession: 2003-067466

Utility

E/ **System and methods using a system-on-a-chip with soft cache**

Inventor: North, Gregory Allen, Austin, TX

Assignee: Cirrus Logic, Inc. 02), Austin, TX

Cirrus Logic Inc (Code: 35367)

Examiner: Bragdon, Reginald G. (Art Unit: 218)

Assistant Examiner: Song, Jasmine

Combined Principal Attorneys: Murphy, Esq., James J.Winstead Sechrest & Minick, P.C.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6622208 | A | 20030916 | US 2001822645 | 20010330 |

Fulltext Word Count: 9219

Description of the Invention:

...ROM) 137 operate from main bus 103 via a local AHB bus 104 and an **interface** 105 which bridges local AHB bus 104 and main AHB bus 103. This configuration minimizes...

...off main AHB bus 103 are a 4-channel DMA engine 106, and flash/SRAM **interface** 107, including an external memory controller, which maps up to 512 MByte external memory into the microprocessor memory space as an extension of on-chip memory, a test **interface** controller (TIC) 108, arbiter 109 and LCD **interface** 110. Test **Interface** Controller (TIC) 108 can take over the bus control from microprocessor 101 and mimic the...

...connected to AHB/APB buses. Arbiter 109 arbitrates bus requests on main bus 103. LCD **interface** 110 supports connections to various LCD panels (since the display may require a large frame...

...An AHB-DSP **interface** 111, which is a slave to main bus 103, allows microprocessor 101 to move data...

...The peripherals operating from APB bus 112 include a USB slave **interface** 114 which supports communications between system 100 and a personal computer (PC) or similar device. When system 100 is used in a portable digital music appliance, this **interface** enables the quick

downloading files from the PC to the portable audio system. UARTa115 is ...

...serial port is fully 16550 compatible and supports various baud rates. It also provides a **legacy** communication channel to an associated PC...

...as Multi-Media Card (MMC). A master mode compatible I2C port 118 provides another common serial **interface** to a range of devices such as EEPROM, DAC/Codecs and some displays...

...GFace 132 interfaces DSP 102 with main bus 102, through slave AHB/DSP **interface** 111, and with the DSP memory. In the illustrated embodiment, DSP 102 is associated with...

...AHB to DSP Slave **interface** 111 allows microprocessor 101 to send read and write requests to the different local memories 133-135 of DSP 102 and global memory 136. Additionally, **interface** 111 synchronizes the microprocessor and DSP clock domains and performs the necessary handshaking. In particular, **interface** 111 responds to transaction requests from the currently granted local or global AHB master 203...

...AHB to DSP slave **interface** 111 operates between two different clock domains. Preferably, the frequency of the DSP clock domain...

...a faster clock. Preferably, a signal is taken from the clock generator which indicates to **Interface** 111 the last DSP cycle before the next microprocessor clock domain rising edge. From this...

...perform byte, half-word and word access to both the RAM and ROM sections. A **wrapper** 138 makes the RAM and ROM AHB compliant slave devices. Since the preferred ROM space is 16-bit wide only, when Microprocessor 101 performs a word-read, **wrapper** 138 issues two consecutive reads to the ROM and concatenates the two read results into ...

...debugging purpose. When TIC mode is enabled, TIC uses 32 pins of the external memory **interface** as a 32-bit bi-directional data bus 301. An external clock (EXTCLKI) is used...

...Display **interface** 110 includes an LCD Display Controller which supports an **interface** to any one of a number of LCD displays. In particular, system 100 can drive...

...FIG. 4 is a functional block diagram of the preferred LCD **interface/display** controller 110. In this embodiment, the display **interface** includes both an AHB bus master 401 and an ...is generated by dividing down the pixel clock as a function of width of the **interface** bus to the external device, and is used internally for such operations as data muxing...

...after receiving an End of Frame signal from Bus Master 401. MCLK is used to **insure** that the display driver voltage frequency does not fall

to DC...

...bus access priority f highest to the lowest as follows: (1) TIC 108; (2) display **interface** 110; (3) DMA controller 106; and (4) Local/Main AHB **Interface** 105...

1/3,K/5 (Item 4 from file: 654) Links
US PAT.FULL.
(c) Format only 2007 Dialog. All rights reserved.

5340267 **IMAGE Available
Derwent Accession: 1999-277009
Utility
REASSIGNED
E/ Integrated customer web station for web based call
management
Inventor: Baker, Thomas E., Monument, CO
Chaffee, Susan L., Colorado Springs, CO
Chen, Yuchien, Colorado Springs, CO
Gruber, Charles J., Monument, CO
Fishman, Howard P., Colorado Springs, CO
Assignee: WorldCom, Inc. 02), Clinton, MS
WorldCom Inc (Code: 48594)
Examiner: Kizou, Hassan (Art Unit: 262)
Assistant Examiner: Pezzlo, John

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6611498 | A | 20030826 | US 98159506 | 19980924 |

Fulltext Word Count: 20410

Description of the Invention:

...3) a data architecture detailing the application, back-end or
legacy data sources available for networkMCI Interact; and...

...pending U.S. patent application U.S. Ser. No. 09/159,695 entitled
INTEGRATED CUSTOMER **INTERFACE** SYSTEM FOR COMMUNICATIONS NETWORK
MANAGEMENT, the disclosure of which is incorporated herein by reference
thereto...

...A third or back-end tier 18 having applications directed to
legacy back-end services including database storage and retrieval
systems and one or more database servers for accessing system resources
from one or more **legacy** hosts...

...in co-pending U.S. patent application Ser. No. 6,115,040, entitled
GRAPHICAL USER **INTERFACE** FOR WEB ENABLED APPLICATIONS, the
disclosure of which is incorporated herein by reference thereto, the
customer workstation includes client software capable of providing a
platform-independent, browser-based, consistent user **interface**
implementing objects programmed to provide a reusable and common
GUI abstraction and problem-domain abstractions. More
specifically, the client-tier software is created and distributed...

...Intranet Dispatcher Server 26; and the MCI Intranet Application servers 30, and the data warehouses, **legacy** systems, etc. 40...

...client applications responsible for presentation and front-end services. Its functions include providing a user **interface** to various MCI services and supporting communications with MCI's Intranet web server cluster 24...

...mentioned, co-pending U.S. patent application Ser. No. 6,115,040 entitled GRAPHICAL USER **INTERFACE** FOR WEB ENABLED APPLICATIONS, the client tier software is responsible for presentation services to the ...The primary common object services include: graphical user **interface** (GUI); communications; printing; user identity, authentication, and entitlements; data import and export; logging and statistics; error...

...after a successful log on. The backplane 12, inter alia, presents a user with an **interface** for networkMCI Interact application management. A typical user display provided by the backplane 12 may...

...may utilize common object services provided by the backplane 12. FIG. 3 shows graphical user **interface** objects 56a,b created and used by a respective application 54a,b for its own...

...FIG. 4 illustrates an example client GUI presented to the client/customer as a browser web page 250 providing, for example, a...

...As shown in FIGS. 3 and 4, the browser resident GUI of the present invention implements a single object, COBackPlane which keeps track of all those...

...to application objects by name. Once retrieved in this manner, the application object's public **interface** may be used directly...

...pass data therebetween is more fully described in the above-referenced, copending application GRAPHICAL USER **INTERFACE** FOR WEB ENABLED APPLICATIONS...

...in co-pending U.S. patent application Ser. No. 09/159,695 entitled SECURE CUSTOMER **INTERFACE** FOR WEB-BASED DATA MANAGEMENT, the contents and disclosure of which are incorporated by reference...

...server. Thus, an application server not only can offer its browser a client to server **interface** through the proxy, but also may offer all its services from its proxy to other...

...include each customer's network management information and data. As shown in FIG. 2, other **legacy** platforms 40(b), 40(c) and 40(d), 610 may communicate individually with the Intranet servers for servicing specific transactions initiated at the client browser. The illustrated **legacy** platforms 40(b)-(d), 610 are illustrative only and it is understood other **legacy** platforms may be interpreted into the network architecture illustrated in FIG. 2 through an intermediate...

- ...All reporting is provided through a Report Requestor **GUI** application **interface** which supports spreadsheet presentation, a variety of graph and chart type presentations, or both simultaneously...
- ...may be maintained to hold the common configuration data which may be used by the **GUI** applications and by the mid-range servers. Such common data includes but are not limited...
- ...maintained by the customer. The call manager webstation 630 includes a web-based graphical user **interface** (**GUI**) application which enables the customers to define their call terminations, and provision routing rules and associated tabular data to control routing by the SCP 610. The **GUI** application also presents alarms and near real time graphical displays of peg counts and ACD...
- ...structures and function profiles may be performed via the call manager webstation's web-based **GUI** application...
- ...Internet connectivity have standard browsers executing Java applets, hereinafter referred to also as a client **GUI** application, downloaded from the web server 632. The web server 632 which is located in...
- ...the network MCI Interact, include Java class files, but no storage of customer data to **insure** data security. Preferably, more than one web server may be provided for redundancy and fail...
- ...As described above, the client webstation 630 provides a web-based graphical user **interface** (**GUI**) offering data management and data presentation features for the call manager system. The web-based front-end **GUI** is typically written using the Java programming language to **insure** platform independence. The client webstation 630 typically includes a web browser with Java applets for the **interface** for providing access to the call manager webstation application from a standard web browser, e...
- ...client webstations) 630 when the Uniform Resource Locator (URL) for the call manager webstation client **GUI** application is accessed...
- ...The call manager webstation client **GUI** application of the system of the present invention is invoked by clicking an icon labeled...
- ...objects shown in FIG. 9. FIG. 9 illustrates the typical objects making up the client **interface** code in one embodiment of the present invention. The user **interface** classes 634 represent the main **GUI** objects for performing call manager specific functionality. Each of the classes, i.e., user and...
- ...FIG. 7 at 640) via the web server 632 are conducted using the common gateway **interface** (CGI). Requests from the client are typically first targeted at a CGI program, which then...
- ...In a preferred embodiment, a Netscape Server Application Program **Interface** (NSAPI) module may be used as an alternative to the CGI

layer, the NSAPI module...described above, the web server 632 provides a communication pass-through between the web client **GUI** application 630 and the back-end call manager integrated data server (CMIDS) 640 which may...

- ...routing engine, e.g., SCP 610. The CMIDS includes databases 642a-c and provides an **interface** to the call manager SCP 610 for rules writing and list management. The CMIDS databases...
- ...as "man machine language" (MML) commands. The CMIDS 640 utilizes MML as well as other **interface** mechanisms supported by the SCP 610. The call manager integrated data server (CMIDS) 640 physically...
- ...670 may reside in the CMIDS, and provide the functionalities described above. The user account **interface** software component 643 generally maintains sessions with the SCPs and provides the functions of the...
- ...handler process generally maintains databases 642a-c and provides reporting facilities. The CMIDS back-end **interface** 712 supports a number of **interface** mechanisms including MML and command line access to the SCP, common alarm and logging services...
- ...Call Manager Client **GUI** Application Implementation...
- ...allows for easy handling of multiple views of a data model. The model is a **wrapper** for an application data object. A controller is a lightweight event handling class, which translates **GUI** events into commands for the application. The view is one particular **GUI** representation of the model. In a MVC typical operation, views register with a model, allowing...
- ...of multiple views when the model changes. Each view has a controller, which handles the **GUI** events, and translates them into command descriptions. The model stores command descriptions, which for example...
- ...webstation application allows authorized customers to manage their ACD data networks via a web-based **interface**. Specifically, customers are enabled to provision hierarchies for their business; control all routing of their...
- ...browser at the webstation 630, deploys a backplane applet via which the call manager client **GUI** applications may be invoked...
- ...conduit through which all other client applications may be deployed, including the call manager webstation **GUI** client application. At step 810, the backplane requests a list of authorized applications from the...
- ...on the call manager icon, triggering the backplane to launch the call manager webstation client **GUI** application...
- ...the back-end returns a list of the available SCP to the front-end web

GUI client application. The proxy generally maintains a "routing engine" list having SCP names and their...

...When the front-end web GUI client application receives the list, a list of routing engine names may be displayed in...For executing the testing process, the debugger/tester uses the MML **interface** to the routing engine, i.e., the debugger/tester formulates the user actions to one...

...a "Backup" option from the administration button menu and invoke the backup functionality. The client GUI application sends a "RTRV-BK-STATUS" message to check the status of the back-end...

...shown in FIG. 15 opens with a list of retrieved gateway types. Typically the client GUI application sends two messages to retrieve information needed to populate the dialog box 980. A...

...the functional areas. For example, strings such as "OK", "Cancel" which are used throughout the GUI, are typically placed in the global list. The class naming convention is "CMXXXStrings" 1044, where...

1/3,K/6 (Item 5 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4854451 **IMAGE Available

Derwent Accession: 2003-438383

Utility

CERTIFICATE OF CORRECTION

E/ **Service installation on a base function and provision of a pass function with a service-free base function semantic**

Inventor: Hunt, Galen C., Bellevue, WA

Assignee: Microsoft Corporation 02), Redmond, WA

Microsoft Corp (Code: 32791)

Examiner: Morse, Gregory (Art Unit: 212)

Assistant Examiner: Das, Chameli C.

Law Firm: Klarquist Sparkman, LLP

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| Main Patent | US 6546553 | A | 20030408 | US 99349732 | 19990708 |
| CIP | Pending | | | US 98197246 | 19981120 |
| CIP | Pending | | | US 98196836 | 19981120 |
| CIP | Pending | | | US 98196974 | 19981120 |

Fulltext Word Count: 17088

Description of the Invention:

...same call signature, including number of arguments and calling convention. Using the same calling convention **insures** that registers are properly preserved and that the call stack is properly aligned between detour...

...A component instantiation function returns a reference to an **interface** of the instantiated component to the client component that called the function. To profile an application, the COIGN instrumentation wraps the **interface** with an **interface wrapper** before returning it to the client component. Subsequent calls to functions on interfaces exposed by the component are intercepted by the **interface wrapper**. Thus, a layer of profiling instrumentation is inserted to measure parameters of function calls...

...installation according to the present invention include timing tests on software components, redirection services for **legacy** support of software, exception handling, and a test harness for a software system...

...keyboards, mice, displays, registries, etc. from any machine in a network. To provide support for **legacy** applications, COP uses the service installation system of the present invention to intercept all application...

1/3,K/7 (Item 6 from file: 654) Links
US PAT.FULL.
(c) Format only 2007 Dialog. All rights reserved.

4780991

Derwent Accession: 2003-197057

Utility

EXPIRED

E/ Integration of objects including Java bytecodes with legacy
3270 applications

Inventor: Gray, James Peyton, Chapel Hill, NC
Kaminsky, David Louis, Chapel Hill, NC
Mathewson, II, James Merwin, Chapel Hill, NC
Peters, Marcia Lambert, Raleigh, NC
Telford, Richard Dean, Cary, NC

Assignee: International Business Machines Corporation 02), Armonk, NY
International Business Machines Corp (Code: 42640)

Examiner: Harrell, Robert B. (Art Unit: 212)

Assistant Examiner: Kang, Paul H

Law Firm: Myers Bigel Sibley & Sajovec

Combined Principal Attorneys: Ray-Yarletts, Jeanine S.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6480895 | A | 20021112 | US 9873423 | 19980506 |

Fulltext Word Count: 12681

Summary of the Invention:

...so widely used, that systems using the 3270 datastream architecture are often referred to as "**legacy**" systems...

...The 3270 datastream architecture was designed before graphical user interfaces (**GUI**) became commonplace. Because of the ease of use that a graphical user **interface** may provide, it is desirable to provide graphical user interfaces for 3270 datastream architecture systems. However, due to the **legacy** nature of 3270 datastream architecture systems, it is generally desirable to provide graphical user interfaces without requiring rewriting of **legacy** code or rearchitecting of **legacy** systems...

...One widely used programming language that can provide a rich graphical user **interface** is Java. As is well known to those having skill in the art, Java programs...

...Accordingly, it would be desirable to integrate Java with **legacy** 3270 applications to obtain the advantages of Java as a portable programming language and the...

- ...Java's centralized application distribution paradigm. However, it would be desirable to integrate Java with **legacy** 3270 applications without the need to write new **legacy** applications or to modify existing **legacy** applications...
- ...Many techniques exist for integrating Java with **legacy** 3270 applications. These techniques include screen-scraping, an HTML gateway, an object request broker and...
- ...host data from one or more sources with new program logic and a graphical user **interface**. The client application gains access to the 3270 data through an application programming **interface** on a terminal emulator. This program can read the screen contents, issue keystrokes and perform...
- ...screen, or it may interpret the data and convert it into a new graphical user **interface** representation. It is even possible to read-in the 3270 screen maps residing on the...
- ...used to replace 3270 datastream communications with object-oriented remote procedure calls. Thus, access to **legacy** data and **legacy** applications may be provided by wrapping them in an object **wrapper**. An Object Request Broker is an example of an object **wrapper**. Unfortunately, this approach may require monolithic new software development. It may not lend itself to incremental replacement of **legacy** applications. However, with the large amount of **legacy** application code that is present, it may not be feasible to replace these applications monolithically...
- ...The above survey indicates that although it is generally desirable to integrate Java with **legacy** 3270 applications, there may be shortcomings with conventional techniques for doing so. Accordingly, there continues...
- ...It is another object of the present invention to integrate Java with **legacy** 3270 applications without requiring rewriting of **legacy** applications...
- ...is yet another object of the present invention to allow incremental integration of Java into **legacy** 3270 applications...
- ...may be placed on the same computer (the host or primary logical unit), where the **legacy** applications reside. The benefit of Java's centralized application administration and automatic software distribution may...
- ...Java bytecodes are transported to the client through the same conduit that transports the 3270 **legacy** data. Thus, new infrastructure need not be developed, and existing management projects, procedures and tools...
- ...used. Java bytecodes including Java applets, images and/or audio may

thereby be integrated with **legacy** applications, both temporally and spatially. Incremental replacement of **legacy** applications with new graphical user **interface**-based Java applications may be provided...

- ...photograph next to the entry. A travel expense reimbursement form can include a graphical user **interface** calculator or a hot link to the company's travel policy that is stored on a different server. Thus, a new Java applet can be used to annotate an existing **legacy** application, without the need to change the **legacy** application itself. It can be used to add more user-friendly help information and/or reduce training costs. Individual panels may be updated to a graphical user **interface** form incrementally...
- ...bytecodes so embedded between the primary logical unit and secondary logical unit, it should be **insured** that the secondary logical unit supports Java. An initialization protocol according to the present invention...
- ...with the other transmissions in the session, such as character-formatted data sent by the **legacy** application. The structured fields of image and audio may include a "Display or Play Now..."
- ...than Java bytecodes. Thus, for example, image, audio and other objects may be integrated with **legacy** 3270 applications by embedding the object in a 3270 datastream structured field, and passing the...
- ...Accordingly, Java and other arbitrary objects may be integrated with **legacy** 3270 applications, without requiring the rewriting of **legacy** applications and allowing the use of highly developed 3270 datastream architecture. Initialization protocols, bootstrapping mechanisms...

1/3,K/8 (Item 7 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4755064 **IMAGE Available

Derwent Accession: 2003-074157

Utility

E/ **System and method for managing a plurality of processor performance states**

Inventor: Cooper, Barnes, Aloha, OR

Assignee: Intel Corporation 02), Santa Clara, CA

Intel Corp (Code: 42458)

Examiner: Ray, Gopal C. (Art Unit: 211)

Law Firm: Schwegman, Lundberg, Woessner & Kluth, P.A.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6457135 | A | 20020924 | US 99371268 | 19990810 |

Fulltext Word Count: 18589

Description of the Invention:

...bootstrapped as normal. For the situation where an ACPI-operating system is installed and a **legacy** solution is enabled, in one embodiment the SMI handler immediately ceases to manage performance state ...support mechanism for performance state transitions consists of the following two software elements: a user **interface** and a device driver. In one embodiment, the user **interface** is an extension of the operating system's configuration/power management application (e.g., the...

...In one embodiment a WDM device driver receives mode commands from the user **interface**. In one embodiment, mode settings are stored in the system registry. The driver registers for...

...When using SMIs to implement performance state transitions in **legacy** operating systems, it is recommended that they be set up on all pertinent events (e...

...as they are received. Once a performance state transition application has been granted the transition **interface**, the application periodically queries the SMI handler for system status. These status calls return information...

...one such embodiment, the application reads the user configuration settings from a control panel user **interface**. By using all of this data, the application can then request that the SMI handler...

...the application. The application can then note the new performance state and update the user **interface**, or retry the operation at a later time if it failed...

...to 170 and SMM support for performance state transitions is disabled, as is the application **interface**.

...

...moves to 174, where SMM support for performance state transitions is disabled and the application **interface** is enabled...disable command. If so, control moves to 180, the application is disabled and the SMI **interface** is enabled...

...SMI handler to cease managing performance state transitions, as well as effectively disabling the SMI **interface** to the performance state transition application. If the application is already loaded, the application receives...

...application loads after the OS has enabled native transition support, the enable performance transition application **interface** will fail the request. In either case, the application will unload once the OS has ...

...routine with the segment registers setup appropriately. In one embodiment this is handled by a **wrapper** routine that performs the segment loading...

...tasks for managing a performance-state-enabled system. The first is that it provides the **interface** to the Performance State Transition Applet for reading status, checking and setting capabilities, and performing...

...The following components are covered in this implementation: the Performance State Transition Applet SMM **Interface** and the State Transition Event Handlers. The Performance State Transition Applet SMM **Interface** routine processes software SMIs from the Performance State Transition Applet, or in response to transition...

...In one embodiment, the Performance State Transition Applet SMM **Interface** is accessed as a hook to the processing of generic software SMIs. In one such...

...values stored in the CPU's general-purpose registers. The next section describes this command **interface** in detail...

...mode, as the OS or applet is in control. When neither the OS nor applet **interface** is in control, the SMM handler can perform performance state transitions in response to AC...

...Windows 95, Windows NT 4.0, Windows 98 and Windows 2000 via an SMM-based **interface**. An applet is provided that communicates to the platform system management interrupt (SMI) handler to...

...one such embodiment, the SMI handler performs all performance state

transitions so that it can **insure** that OS registers are properly saved/restored in order to prevent collision with accesses by...

...are applied to operating systems other than Windows 98 and Windows 2000. This is because **legacy** non-ACPI compliant operating systems used CPU speed dependent calibration loops for driver timing routines...

...3) SMI+Applet (**legacy** non-ACPI compliant operating systems). A one-time transition from high to low frequency is...

...The second SMI command port value is used for the performance state transition applet **interface**. In this case, in order to claim only one additional SMI command port value and...

...To keep the **interface** isolated from collisions from ACPI OS commands, in one embodiment the **interface** uses only one register to initiate the SMIs. In addition, returned data is reported back...

...In one embodiment, the following commands can be used to communicate with the SMM **interface**: Performance State Disable, Performance State Control, Get Performance State Status, Set Performance State, and Set...

...this command has two subfunctions that allow the applet to enable and disable the SMM **interface**. However, the Get Performance State Status function is always enabled...

...which setup mode the user has selected, AC status, and the revision of the SMM **Interface** it is compliant with. (NOTE: This function must maintain memory copies of all variables to...

...user BIOS setup options to reflect changes made by the user in the applet user **interface**. The setup options include disabling performance state transitions, operating in one particular state all of

...

1/3,K/9 (Item 8 from file: 654) Links
US PAT.FULL.
(c) Format only 2007 Dialog. All rights reserved.

4739012 **IMAGE Available
Derwent Accession: 2002-705418
Utility
CERTIFICATE OF CORRECTION
E/ Thermal control within systems having multiple CPU
performance states
Inventor: Cooper, Barnes, Aloha, OR
Assignee: Intel Corporation 02), Santa Clara, CA
Intel Corp (Code: 42458)
Examiner: Ray, Gopal C. (Art Unit: 211)
Law Firm: Schwegman, Lundberg, Woessner & Kluth, P.A.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6442700 | A | 20020827 | US 99371381 | 19990810 |

Fulltext Word Count: 21249

Description of the Invention:

...bootstrapped as normal. For the situation where an ACPI-operating system is installed and a **legacy** solution is enabled, in one embodiment the SMI handler immediately ceases to manage performance state ...support mechanism for performance state transitions consists of the following two software elements: a user **interface** and a device driver. In one embodiment, the user **interface** is an extension of the operating system's configuration/power management application (e.g., the...

...In one embodiment a WDM device driver receives mode commands from the user **interface**. In one embodiment; mode settings are stored in the system registry. The driver registers for...

...When using SMIs to implement performance state transitions in **legacy** operating systems, it is recommended that they be set up on all pertinent events (e...as they are received. Once a performance state transition application has been granted the transition **interface**, the application periodically queries the SMI handler for system status. These status calls return information...

...one such embodiment, the application reads the user configuration settings from a control panel user **interface**. By using all of this data, the application can then request that the SMI handler...

...the application. The application can then note the new performance state and update the user **interface**, or retry the operation at a later time if it failed...

...to 170 and SMM support for performance state transitions is disabled, as is the application **interface**.

...

...moves to 174, where SMM support for performance state transitions is disabled and the application **interface** is enabled...

...disable command. If so, control moves to 180, the application is disabled and the SMI **interface** is enabled...

...SMI handler to cease managing performance state transitions, as well as effectively disabling the SMI **interface** to the performance state transition application. If the application is already loaded, the application receives...

...application loads after the OS has enabled native transition support, the enable performance transition application **interface** will fail the request. In either case, the application will unload once the OS has

...

...routine with the segment registers setup appropriately. In one embodiment this is handled by a **wrapper** routine that performs the segment loading...

...tasks for managing a performance-state-enabled system. The first is that it provides the **interface** to the Performance State Transition Applet for reading status, checking and setting capabilities, and performing...

...The following components are covered in this implementation: the Performance State Transition Applet SMM **Interface** and the State Transition Event Handlers. The Performance State Transition Applet SMM **Interface** routine processes software SMIs from the Performance State Transition Applet, or in response to transition...

...In one embodiment, the Performance State Transition Applet SMM **Interface** is accessed as a hook to the processing of generic software SMIs. In one such...

...values stored in the CPU's general-purpose registers. The next section describes this command **interface** in detail...

...mode, as the OS or applet is in control. When neither the OS nor applet **interface** is in control, the SMM handler can perform performance state transitions in response to AC...

...Windows 95, Windows NT 4.0, Windows 98 and Windows 2000 via an SMM-based **interface**. An applet is provided that communicates to the platform system management interrupt (SMI) handler to...

...one such embodiment, the SMI handler performs all performance state transitions so that it can **insure** that OS registers are properly saved/restored in order to prevent collision with accesses by...

...are applied to operating systems other than Windows 98 and Windows 2000. This is because **legacy** non-ACPI compliant operating systems used CPU speed dependent calibration loops for driver timing routines...

...3) SMI+Applet (**legacy** non-ACPI compliant operating systems). A one-time transition from high to low frequency is...

...The second SMI command port value is used for the performance state transition applet **interface**. In this case, in order to claim only one additional SMI command port value and...

...To keep the **interface** isolated from collisions from ACPI OS commands, in one embodiment the **interface** uses only one register to initiate the SMIs. In addition, returned data is reported back...

...In one embodiment, the following commands can be used to communicate with the SMM **interface**: Performance State Disable, Performance State Control, Get Performance State Status, Set Performance State, and Set...

...this command has two subfunctions that allow the applet to enable and disable the SMM **interface**. However, the Get Performance State Status function is always enabled...

...which setup mode the user has selected, AC status, and the revision of the SMM **Interface** it is compliant with. (NOTE: This function must maintain memory copies of all variables to...

...user BIOS setup options to reflect changes made by the user in the applet user **interface**. The setup options include disabling performance state transitions, operating in one particular state all of

1/3,K/10 (Item 9 from file: 654) Links
US PAT.FULL.
(c) Format only 2007 Dialog. All rights reserved.

4712115 **IMAGE Available
Derwent Accession: 2001-191263
Utility
EXPIRED

E/ **Braille computer monitor**

Inventor: Becker, John V., late of New York, NY, deceased
Becker, David A., 87 Remsen St., Brooklyn, NY, 11201, legal
representative
Hinton, Daniel E., 815 Galway Garth, Arnold, MD, 21012-1342
Anderson, Jr., Hugh G., 1755 Manchester Rd., Westminster, MD,
21157

Assignee: Unassigned
Unassigned Or Assigned To Individual (Code: 68000)

Examiner: Liang, Regina (Art Unit: 264)

Law Firm: Katten Muchin Zavis Roseman

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6417821 | A | 20020709 | US 2000605514 | 20000628 |

Fulltext Word Count: 5833

Description of the Invention:

...belt power transmission system 800, and a support structure 900. Not shown are a computer **interface**, an outer casing, a computer and an electronic control package of generally known arrangement...

...100 have a spring that returns them to the raised position. The twice tapping requirement **insures** intentional versus accidental communicative desires and is similar to double clicking with a mouse...

...deflections imposed by dot pin drag forces. Part of the difficulty was from working with **legacy** dot pin dimensions. Therefore, in configuring an erasing mechanisms 400 especially for multi-line TCMs...

...The user **interface** for the erasing mechanism 400 is a hand operated electronic switch that is surface mounted...

...A standard computer **interface** is connector-mounted on the traveling printhead 300 and connects the TCM to a standard...

...wired to the connector. The outer casing of the TCM is a conformal shell or **wrapper** that provides a smooth **interface** for the user while preventing the user from exposure to the inner workings of the...

1/3,K/11 (Item 10 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4652243 **IMAGE Available

Derwent Accession: 2000-064334

Utility

E/ Method for computer internet remote management of a telecommunication network element

Inventor: Barker, William E., Geneva, IL
Connelly, Lisa M., Lisle, IL
Eggert, Marvin A., Aurora, IL
Foley, Michael P., Elmwood Park, IL
Macfarlane, Kenneth R., Wheaton, IL
Parsons, Philip M., Lisle, IL
Rai, Girish, Bartlett, IL
Rog, Jerome E., Bartlett, IL
Vangsness, Kurt A., Aurora, IL

Assignee: Lucent Technologies, Inc. 02), Murray Hill, NJ
Lucent Technologies Inc (Code: 39644)

Examiner: Rinehart, Mark H. (Art Unit: 212)

Assistant Examiner: Kang, Paul

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6363421 | A | 20020326 | US 9888463 | 19980531 |

Fulltext Word Count: 21821

Description of the Invention:

...The client executes the Client **Interface** and propriety applications via Web pages. Microsoft Internet Explorer and netscape browsers are supported as...

...the web-enabling devices for PCs and X-terminals. Through a Web-based graphical client **interface**, clients' commands generate HTTP requests to the element management system server. The server gathers information...

...network element detailed status display. Client applications communicate with the server via an object oriented **interface** to the element manager API (EMAPI) 55 through the distributed object request architecture (CORBA) 48. This **interface** provides a consistent **interface** to all managed objects in the network, and hides the implementation details associated with the...

...The element management system client 28 is the client's **interface** to the element management system server 32. It consists of the web

browser 45 and...

- ...Web Browser: The web browser 45 is the **interface** to end client, a host for JAVA applets 44, and a virtual machine for JAVA...
- ...Command Line Interpreter 76: provides an ASCII command language **interface** to allow the technician to enter commands at the element management system and observe results...
- ...UX Proxy 78: UX message **interface** (bridge between System V message queues and sockets) to an internal database subsystem (IDS) 79...
- ...Command Handler 88: Handles AP administration command requests issued from either the **GUI** based or text based client interfaces. Executes a RAP 90 to complete the administration request...
- ...Text Command Interpreter 92: A text based **interface** for special situations in which the **GUI** based **interface** presented by the element management system Server is not available to the client...
- ...RAPs 90: The Resource Administration Process is an application processing **interface**(API) for fork-exec'ing a process and obtaining the results of the process execution...
- ...command and control and fault management. These web browser hosted applications provide a graphical client **interface** based client **interface** in a cross ...The client **interface** to the server is described in the EMAPI 55 described in a pending patent application...
- ...by reference herein. The EMAPI 55 is implemented utilizing an industry standard object management group **interface** description language (IDL). The interfaces and semantics of the EMAPI 55 enable client application processes to utilize this **interface** to provide management of the system. Distribution of this **interface** is achieved through use of the Common Object Request Broker Architecture (CORBA) which provides a...
- ...The development of client applications depends only on the EMAPI 55 **interface** specification. The use of CORBA allows the clients to be distributed and implemented in a...
- ...any other system aspects that are not part of an object's **interface**
...
- ...ovspmd (OpenView System Process Management Daemon): This process must be running to enable client-**interface** status-checking programs such as ovstart to work. Since this monitor does not support restarting...
- ...SNMP Collection Daemon which allows clients to define, via the HP OpenView Windows X-based **GUI interface**, SNMP MIB values that are to be collected periodically. It provides ways to define thresholds...

- ...ovw (OpenView Windows): The OpenView Windows X-based **GUI** provides access to map applications, an event browser, and a MIB browser...
- ...The HPOV SNMP API, a C-language **interface** to this runtime system, is provided as part of the HPOVNNM Developer's Kit and...
- ...Provides UX messaging **interface** to the IDS 79 to receive and process IDS triggers. A trigger is needed to...
- ...single message type for forwarding all element management system-generated TI/OP messages. The current **interface** from a mobile switch center (MSC) ECP ROP supports specifying Alarm Level (MAN, INFO, CRIT...
- ...The client **interface** to the services and the managed object attributes and methods is described in the EMAPI...
- ...know whether the underlying protocol to the network element is SNMP, CMIP or a proprietary **interface**). Managed object specific logic is encapsulated within the managed object instead of scattered throughout various...
- ...The definition of these managed object class identifiers and attribute codes is part of the **interface** definition between all service objects and their clients...
- ...interfaces on the AP. They both provide common Ethernet functionality but there is an ethernet **interface** node (EIN) and a LAN managed object...
- ...especially when there are multiple interfaces to it within the server) by placing it within **wrapper** libraries. For example, the Element management system Logger class will provide object-oriented wrappers for ...
- ...**Interface** to clients for starting a session...
- ...**Interface** to clients for manually ending a session...
- ...**Interface** to clients for periodic check-in (heartbeat...
- ...**Interface** to other server components for registering interest in notifications of session/application termination. The components...
- ...Notification to registered entities when a session/application has been ended via the callback notification **interface** described above...
- ...be generated by element management system components on the same machine, or even by other **legacy** network elements. The Event Distributor may be implemented as a set of objects within the...
- ...for the purpose of simple, open-ended event correlation. The Event Screener supports the same **interface** (although not available to clients) as the Event Distributor, but is only for use within...

- ...Provide an IDL **interface** (Event Distributor Only) for registering filters based on the following...
- ...Provide an IDL **interface** for clients (Event Distributor only) and other Object Server components to explicitly cancel a specific...Object Server **Interface**
- ...
- ...Provide an **interface** to managed object classes in the object server to support...
- ...SNMP **Interface**
- ...
- ...The SNMP Mediator handles all interfacing with SNMP agents on network elements. The SNMP **interface** consists of Attribute Polling, Configuration Auditing, Command Execution, SNMP Retry Mechanism, and Trap Delivery...
- ...of sequence, a command request/response convention between manager and agent will be utilized to **insure** that an agent will respond only once to a single command (i.e. SET operation...
- ...be consistent for all GET, GET-BULK and SET operations and will conform to the **interface** prescribed by the HPOVNNM SNMP library or the CMU SNMP library. For further information, see...
- ...be identified through a class dictionary containing attribute codes and type information, available via common **interface** definition to both server and client code. For example, the AP object may contain attributes...
- ...within the element management system server and not by a network element through the SNMP **interface**. The element management system server must generate the command acknowledgment and command response events itself...
- ...Provides **interface** to clients for attribute registration given the following parameters...
- ...Provides **interface** with client for de-registration...
- ...Provides **interface** with Client Session Manager for de-registration when abnormal client termination is detected via audit...
- ...Client **Interface** Components...
- ...components present on a client of the element management system server to provide the client **interface** to NE management. Specific client **interface** style and content will be addressed after the architecture with input from human factors and element management system engineering. In addition to making the client **interface** as easy to use as possible, the client **interface** must retain similarities

with the current maintenance model such that little retraining of the end
...

...The primary client **interface** is provided by an HTML web browser.
Both Netscape navigator and Microsoft Internet Explorer are...
...Java-based **GUI** Infrastructure...

...This section describes base components that are necessary for
implementing the AP specific **GUI** applets. A number of these
components (especially the **GUI** components) may be satisfied (or
based upon) commercial 3rd party products (for example Rogue Wave
JWidgets, or Microline's Grid widget). Also, 3rd party non
GUI container and algorithm classes (either Rogue Wave or JGL for
example) should be considered to...context sensitive command execution
through the use of pull down (or pop up) menus. The **interface** for
command execution and display of command results is the same as the
interface described in the "Command Handler" section above...

...This application provides an alarm browser

interface to active
alarms within the system. The client can specify a filter to limit the...

...Each managed object class will adhere to the **interface** specified
by the managed object base class (provide for client attribute
registration, notification, configuration registration...

...the element management system server is back up. The definitive
requirements for element management system **GUI** Client version
management will be described in the element management system **GUI**
Client Capability Requirements/High-Level Design document...

...MIB is that it contains a very detailed description of the element
management system/AP **interface**. The MIB is intended to serve as
much as possible as a single element management system/AP
interface definition. As such, its details may need to be modified
more frequently than at each...

...with global access permissions for use by infrastructure components
which make use of the same **interface** definition...

...SNMP Agent: Provides the **interface** to the element management
system Server using the SNMP protocol and a MIB defined specifically...

...NE Status API: an **interface** for writing to and reading from the
Network Element Status Table...

...Command/Response API: **interface** between a command source and the
Command Handler for the purpose of issuing commands and...

...ECP Agent: **interface** to Status Display on the ECP;
interface to IDS-AP for processing triggers indicating change in
data. Passes triggers on to interested...

- ...interfaces to the AP via the SNMP Agent. A MIB is used to define the **interface** between the element management system Server and the Agent and is common to both the...
- ...is used by the Manager to determine whether this is an MIB (supporting the other **interface** conventions described in this section), and also to provide a versioning mechanism to support MIB...server infrastructure required to support object services. The following is the Element Management Application Programming **Interface** (EMAPI) 55 in accordance with the invention utilized by EM Clients...
- ...request brokers (ORBs). Application services are provided through object interfaces formally defined in the CORBA **Interface** Definition Language (IDL...
- ...a session. Application identifiers are assigned by the Client. For the Element Manager Graphical User **Interface**, each "window" will be assigned a unique application id. Note that each Client is required...
- ...Each managed object service class must implement the MO **interface**, which defines the following configuration and status services...
- ...client may use this method to register for a snapshot of current status information. This **interface** differs from the previous one in that the requested attribute list may specify any managed...
- ...Each network-element level managed object must also implement the NEMO **interface** which defines additional network-element level configuration services...

1/3,K/12 (Item 11 from file: 654) Links
US PAT.FULL.
(c) Format only 2007 Dialog. All rights reserved.

4639510 **IMAGE Available
Derwent Accession: 2002-205301
Utility

E/ Computer software for converting a general purpose computer
network into an interactive communications system

Inventor: Simonoff, Adam J., Fairfax Station, VA

Assignee: The United States of America as represented by the Secretary of
the Navy 06), Washington, DC

U S of America Navy Secretary of (Code: 86584)

Examiner: Lim, Krisna (Art Unit: 213)

Combined Principal Attorneys: Bechtel, Esq., James B.; Powell, Jr., Esq.,
Raymond

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6351777 | A | 20020226 | US 99296746 | 19990423 |

Fulltext Word Count: 16270

Description of the Invention:

...storage device such as a hard disk and a communications device,
e.g., a network **interface** card. It should also be mentioned that
computers 300a-300r can include desktop computers, laptop...

...that the present invention was developed in response to perceived
problems in the interoperability of **legacy** computer hardware used
in combat systems and networks and solved those problems. However, since
the...

...of the windows for the client-side White Board display, i.e., the White
Board **GUI** will be presented to the user. The user can then run the
White Board application...

...the White Board displayed on the computers 300a and 300b employ an
intuitive graphical user **interface** (**GUI**). Moreover, each
user is provided with information regarding his respective White Board.
For example, White...selection of an object from the resource list
depicted in FIG. 7 creates an empty **wrapper**, which **wrapper**
is assigned a unique identifier and which **wrapper** contains the
selected object label. Thus, when a mouse down event occurs at step S21

...
...S24, the information needed to regenerate the selected object is placed
into the above-mentioned **wrapper**, to thereby generate a

wrapper object. As mentioned above, the **wrapper** includes a unique identifier so that the **wrapper** object can be locally identified, used by the local White Board client 301 and globally identified to prevent collisions with other **wrapper** objects. During step S26, the **wrapper** object is added to a vector holding all **wrapper** objects drawn on the local White Board client 301. Using the thus generated vector, the **wrapper** object is displayed in the White Board client 301. When a "mouse up" event occurs, the **wrapper** object is transmitted to the White Board server 102 over LAN 400 for relay to...

...up" should be understood to equate to "hard return" with respect to text, since the **wrapper** object containing text is transmitted to White Board server 102 when a hard return is...

...i.e., both White Board client and White Board server, keeps track of all the **wrapper** objects to be displayed in their respective order via a vector. The vector advantageously stores...

...each object, i.e., each object in the hash table is referred to via a **wrapper** object. Thus, the **wrapper** object tells the White Board the kind of object to display, its location, size and...

...The **wrapper** object provides for an open architecture design so that developing new objects for use with the White Board is greatly simplified. Stated another way, the **wrapper** allows third party objects to simply plug-in to the White Board. The **wrapper** advantageously can be used to define additional characteristics of an object to be displayed without...

...a part of the Java runtime environment. In contrast, optional signifies that a White Board **interface** class may be implemented which requires certain subroutines or methods be available in the source...

...White Board system was developed to permit the White Board server to filter the data, **wrapper** objects, by privilege. Additionally, in order to **insure** traceability, i.e., the ability to retrace or recreate the steps by which the White...

...recreate the same object remotely. In other words, the White Board server time stamps each **wrapper** object so that the White Board system can afterwards determine when the **wrapper** object was created and when the **wrapper** object was modified, and stores a copy of the **wrapper** object on the White Board server (or at a White Board server specified location). Given...

...each user's privilege before deciding whether or not that particular user will receive a **wrapper** object. A more efficient method of relaying **wrapper** objects advantageously can be implemented when all users are equal in privilege. In that case...

...the White Board contents can be changed simultaneously by multiple users with one exception. The **wrapper** around each object allows the White Board system to establish read/write privileges on objects...

...As discussed above, the global key name associated with each **wrapper** object contains information regarding the user and White Board client that created that **wrapper** object; the object's **wrapper** also contains the name of the last user to modify the object. Moreover, the White Board server time stamps every change, i.e., every **wrapper** object, as it arrives from one of the White Board clients or as it is...

...Moreover, as previously discussed, each **wrapper** object, e.g., each object generated by White Board client 301, derives its unique identifier...

...A second Hash Table (named HashTable) contains the object's **Wrapper** associated with a global key name known by ALL privileged White Board clients on the...

1/3,K/13 (Item 12 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4206909 **IMAGE Available

Derwent Accession: 1997-559168

Utility

E/ **System to transition an enterprise to a distributed infrastructure**

Inventor: Eager, Timothy, Fullerton, CA

Anand, Madhav, Cambridge, MA

Aslanian, Edouard, Hermosa Beach, CA

Assignee: i-CUBE 02), Cambridge, MA

i CUBE

Examiner: Hafiz, Tariq R. (Art Unit: 272)

Assistant Examiner: Dam, Tuan Q.

Law Firm: Hamilton, Brook, Smith & Reynolds, P.C.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 5960200 | A | 19990928 | US 96714205 | 19960916 |
| Provisional | | | | US 60-16330 | 19960503 |

Fulltext Word Count: 19684

Description of the Invention:

...inter-connected through communication links 170. The data storage layer 150 preferably includes a user **interface** repository 152, a business process repository 154, a business object repository 156, and a data...

...A preferred re-architecting system 20 includes a user **interface** conversion utility 210, a procedural language conversion utility 220, and a data definition language conversion...

...130 and the data access layer 140 of the multi-tier architecture 10. The user **interface** conversion utility 210 is in communication with the user **interface** repository 152 and the data definition language conversion utility 230 is in communication with the...

...A preferred re-engineering system includes a graphical user **interface** editor 310, a graphical business process editor 320, a graphical business object editor 330, a...

...communication with the functionality layer 130 of the multi-tier architecture 10. The graphical user **interface** editor 310, the graphical business process editor 320, the graphical business object

editor 330, and the graphical data record editor 340 are in communication with the user **interface** repository 152, the business process repository 154, the business object repository 156, and the data...

...the architecture 10 supports both custom-developed applications as well as application converted from a **legacy** system. The IMS/VS **legacy** environment is used herein to illustrate, but not limit, architectural concepts that pertain to a converted **legacy** application...

...2 is a functional block diagram of the interrelationships of FIG. 1. Conceptually, the user **interface** translator 210 and the graphic user **interface** editor 310 affect the presentation layer 110 of the multi-tiered architecture 10. The graphical...42 through to the operation stage 48, the amount of operations support required by the **legacy** system decreases and the amount of operations support for the open system increases, as illustrated...

...and performance issues that arise when attempting substantial changes to an application designed for a **legacy** system and converted to a multi-tiered client/server architecture. Custom development refers to the ...

...of support staff for the maintenance phase. One powerful example of integration at the user **interface** layer using the OSI process 469 is the creation of a corporate intranet using internet...

...such as Java from Sun Microsystems to provide a user-friendly, platform independent, common user **interface** to corporate application...

...also take the form of an X-terminal, a workstation console, or a Macintosh style **interface** display. As shown, the presentation layer 110 includes a processor 111 having the current screen...

...also includes internal or external storage, such as a disk device, from which a user **interface** engine is loaded into the memory of the processor 111 as required. For a personal...

...As shown in FIG. 4, the presentation layer 110 includes a user **interface** display platform 115, an application user **interface** representation mechanism 116, and a user **interface** engine 117. In a preferred embodiment of the present invention, the user **interface** display platform 115 is a conventional Graphical User **Interface** (GUI) tool, commercially available. Consequently, the user **interface** display platform 115 has its own internal user **interface** representation mechanism 118 to display the various components of a user **interface**, usually in a graphical way...

...Preferably, the underlying internal user **interface** of the user **interface** display platforms 115 is preferably derived from a frame-based system. A frame system is...

...Similarly to the display platform user **interface** representation

structures 118, the application user **interface** representation structures 116 store descriptive information representative of the different objects that compose a user **interface**. Each object is described by a structure comprising a plurality of fields containing information representing...

...attribute of that object or a relationship between the object and another object. The user **interface** engine 117 maps each of the different objects that compose the user **interface** of a given application into the corresponding representations 118 in the user **interface** display platform 115 of choice for that application...

...On the one hand, the user **interface** engine 117 requests application user **interface** representation structures 116 from the business process layer 120. Once the business process layer 120 satisfies the request, the user **interface** engine 117 converts the application user **interface** representation structures 116 just received into user **interface** representation structures 118 that are expected by the user **interface** display platform 115 for display to the end user on a display station 111...

...through the display station 111, such as selecting an item or modifying information, the user **interface** engine 117 translates that user request from user **interface** display platform representation structures 118 into the corresponding application user **interface** representation structures 116, which are then handed to the business process layer 120 for execution...

...FIG. 5 is a schematic diagram of a sample mapping between application user **interface** representation structures 116 and display platform user **interface** representation structures 118. In the figure, the user **interface** display platform 115 is exemplified as Microsoft Windows 3.x and the display platform user **interface** representation structures 117 are thus the internal Windows 3.x management structures. However, other user **interface** display platforms 115 using similar internal structures to manage windows are supported by the exact same user **interface** engine 117. Notably, the internet's world-wide web, based on the HTML or Java user **interface** languages, is another example of user **interface** display platform 115. Indeed, in a preferred embodiment of the present invention, the user **interface** engine 117 is written using Microsoft Visual C++ and based on the industry-standard Microsoft...

...platform development for Windows 3.x, Windows 95, Windows NT, MacOS, and UNIX-based user **interface** display platforms 115, including internet web servers...

...FIG. 6 is a block diagram of the operational modules of the user **interface** engine 117 of FIG. 4. The user **interface** engine 117 includes an initialization module 117-1, a user input module 117-2, and...

...During initialization, the user **interface** engine 117 first initializes its initial state, setting up any structures necessary for operation. Depending on the implementation, the user **interface** engine 117 can then initialize communications with the business process layer 120, receiving a client identification number. Depending on the implementation, the user **interface** engine 117 can also display an initial application menu or screen, initial objects that are...

...After completing the initialization, the user **interface** engine 117 continues to the user input module 117-2. The user **interface** engine 117 waits for user input and processes it accordingly. In particular, the user input module 117-2 handles interactions with GUI objects and performs application-dependent actions in response to user inputs...

...router communications module 117-3. In the state router communications module 117-3, the user **interface** engine 117 creates outgoing application user **interface** representation structures 116 from the screen data and packs these structures for delivery to the business process layer 120. Typically, the outgoing application user **interface** representation structures 116 contain values of screen fields which have changed since the previous call to the business process layer 120. The packed application user **interface** representation structures 116 are then sent to the business process layer 120, which returns packed application user **interface** representation structure 116 describing the result of the transaction. The packed application user **interface** representation structures 116 returned from the business process layer 120 are then unpacked and processed...

...long as the business process layer 120 does not indicate a fatal error, the user **interface** engine 117 processing continues (resumes the wait for user input) at the user input module...

...Most of the user **interface** engine 117 processing occurs in the handling of screens: building a screen from a description...

...more specifically its main state router component (described below), is always initiated by the user **interface** engine 117 because a remote procedure call (RPC) mechanism which interfaces the user **interface** engine 117 with the business process layer 120 is preferably unidirectional and synchronous...

...To simulate asynchronous communication using a unidirectional synchronous RPC model, the user

interface engine 117 includes an ability to periodically poll the state router for messages during the user **interface** engine's 117 idle time, namely when there is no user input to be processed...

...Essentially, during idle message polling the user **interface** engine 117 queries the state router for any initial messages. At the start of an...

...logon screen, application menu, and other object for the user to act upon, the user **interface** engine 117 waits to process user inputs. If the user takes no action and idle message polling is enabled, the user **interface** engine 117 will periodically query the state router for any messages. If message polling is...

...structure, which is preferably a two-way associative array, it is possible for the user **interface** engine 117 to allow window control handlers of the user **interface** display platform 111 to manage general window operation and make callbacks to the user **interface** engine handlers when an action is required, for example, when a button is pressed...

...In a preferred embodiment of the present invention, the user **interface** engine 117 can process any type of action from any type of screen object, e...

...pressed. Typically, when an action is performed, one of two things may happen: the user **interface** engine 117 performs some internal function based on the action, or sends information to be...

...access server runtime environment resides, and a terminal console 136 which serves as a human **interface** for host administration purposes. In addition, a communications controller 138 such as a LAN controller, modem or similar device serves as an **interface** to a communication link. The host computer system 132 can be considered conventional in design...

...is a state router 122. Conceptually, the state router 122 receives requests from the user **interface** engine 117 (FIG. 4) and, based on the request, determines which actions to take. The...

...the state router 122 accepts any resulting return information and forwards it to the user **interface** engine 117...

...The requests received from the user **interface** engine 117 include application user **interface** representation structures 121. The application user **interface** representation structures 121 include request identifiers, transaction codes, screen information, and input/output buffers. A...

...be executed in response to the request. There is one request identifier for any user **interface** event caused by the user. In this regard, request functions are similar to the conventional callbacks found in GUI languages such as X-Windows developed at the Massachusetts Institute of Technology, in Cambridge, Mass...

...FIG. 9 is a flow diagram of the communication mechanism between the user **interface** engine 117 and the state router 122. As depicted, the user **interface** engine 117 includes a user **interface** routine 117-6 and initiates the communication by calling a pass message function 117-8. The pass message function 117-8 first compresses the application

user **interface** representation structures 116 to be transmitted into a single request string using a packing procedure...

...request string compression performed by the packing procedure is necessary because the outgoing application user **interface** representation structures 116 cannot be transferred efficiently as such across the communication link...

...routine 117-9 takes two parameters: the request string to be passed from the user **interface** engine 117 to the state router 122 and the return string to be returned to the user **interface** engine 117 from the state router 122. From the point of view of the state...

...The unpacking procedure converts the request string into an array of request application user **interface** representation structures 121. This array is then passed to a main state router 122-1...

...Once the state router 122 completes its processing, the resulting array of return application user **interface** representation structures 121 is again packed into a return string, which is passed back to the user **interface** engine 117 using an RPC mechanism 122-9...

...Because new application user **interface** representation structures 121 can be added to facilitate the transport of new types of objects... (e.g., bit integers), and strings (both variable- and fixed-length). To create a new application user **interface** representation structure 121, a developer need only create packing and unpacking routines for that structure...

...under ANSI C) or using object classes (under ANSI C++). Although the ANSI C language **interface** is very usable, the ANSI C++ language **interface** makes use of object-oriented features such as virtual functions to make packing and unpacking...

...level packing and unpacking routines take arrays (or, in ANSI C++, containers) of application user **interface** representation structures 121 and create a single character string containing the packed information suitable for...

...contains type information as well as member data, so that any sequence of application user **interface** representation structures 121 can be sent and properly reconstructed at the receiving end...

...router 122 of FIG. 8. Initially, when a logon request is received from the user **interface** engine 117 through the request user **interface** structure 121a and then authorized, the state router's 122 internal state is initialized with...loaded from the database repository 152. This screen information is passed back to the user **interface** engine 117 through the return application user **interface** representation structures 121. The user **interface** engine 117 then displays the screen and awaits user input. When a user enters or changes data on a screen and presses a function key, the user **interface** engine 117 translates this user input into a request to

the state router 122...

...and the current state is then updated with any new field values from the user **interface** engine 117. State information is represented by a field state structure 122b. Then, the state...

...which describes a function to be executed, is included with the request from the user **interface** engine 117. The state router 122 verifies the user's authorization to perform this function...

...from the database repository 152, and its information is included in the return application user **interface** representation structures 121b destined for the user **interface** engine 117. The state router 122 then recalls the functionality server that corresponds to the...

...its value. The new values and attributes are then included in the return application user **interface** representation structures 121 array passed back to the RPC mechanism for return to the user **interface** engine 117...

...of custom functionality servers, the state router 122 still processes requests received from the user **interface** engine 117 in response to user **interface** event caused by the user in manner similar to that described earlier. However, much of...

...redirects processing to appropriate functionality servers based on the transaction codes received from the user **interface** engine 117. However, when the functionality server returns, it passes back an event to the...

...data server exists for each of the four application object repositories. Consequently, there is user **interface** data server 141 to manipulate user **interface** objects 142, a business process data server 143 for business processes 144, a business object...

...a database server 147 for application data records 148. The data servers constitute the sole **interface** between the data storage layer 140 and the functionality layer 130, and each data server...

...multiple clients. The set of functions, or services, provided by a server constitutes the server **interface**. This **interface** is specified in an **Interface** Definition Language (IDL) file. The concept of servers is well known, and details of server...

...the present invention, this query string is forwarded to the database using the Oracle Call **Interface** (OCI) from Oracle Corporation. At a high-level, the process consists in initializing bind and...

...150 is a repository for data accessed by the data access layer 140. The user **interface** data repository 152 provides user **interface** objects 153 to the data access layer 140. The business process data repository 154 provides...

...which all database data resides, and a terminal console 155 which serves

as a human **interface** for host administration purposes. DBMS log files are stored in storage unit. A printer 157...

...back, with appropriate error messages posted. Transaction management 161 is useful in distributed systems to **insure** data consistency in the absence of user-defined integrity constraints...

...Network management 166 provides a graphical **interface** to monitor clients, servers, and brokers. Network traffic and performance can thus be monitored, and...

...application version management functions are provided. In addition, currency is handled through locking functions to **insure** data consistency. Data integrity is controllable at the functionality layer by the business objects rules...Returning to FIG. 1, the re-architecting system 20 includes a user **interface** conversion utility 210, a procedural language conversion utility 220, and a data definition language conversion...

...FIG. 16 is a block diagram of the user **interface** conversion utility 210 of FIG. 1. The user **interface** conversion utility 210 converts the user **interface** of an existing application represented by the source user **interface** definitions 211 into target user **interface** definitions 213 using the user **interface** converter 212. In a preferred embodiment of the present invention, the source user **interface** definitions 211 can be viewed as IMS/VS Message Format Service (MFS) files...

...Target user **interface** definitions 213 can take one of three forms: database files 246, a header file 247, and a

GUI file 248.

Database files 246 contain the set of statements necessary to populate user **interface** repository 152 with screen and message information for MFS file 211. In a preferred embodiment...

...A deletion script removes from the user **interface** repository 152 any definitions for the MFS file 211. Once this repository cleanup is accomplished, an insertion script adds to the user **interface** repository 152 any new definitions for the MFS file 211. Consequently, the user **interface** conversion utility 210 can be run multiple times for the same MFS file without negative effects. In a preferred embodiment of the present invention, the user **interface** repository 152 is a standard RDBMS such as Oracle Server 7 from Oracle Corporation

...Information stored in the user **interface** repository 152 is converted at application runtime into the user **interface** representation structures of the presentation layer 116. The user **interface** engine 117 of the presentation layer 110 then maps application user **interface** representation structures 116 into display platform user **interface** representation structures 118, used by the user **interface** display platform 115 for display to the user. Accordingly, target user **interface** definitions 213

effectively constitute an intermediary user **interface** definition language for storage of user **interface** information in the user **interface** repository 152 and eventual user **interface** representation structures 118...

...an alternative to database files 246. In a preferred embodiment of the present invention, user **interface** representations are stored in the user **interface** repository 152 and retrieved as needed from this repository by the business process layer 120. This is an appropriate mode of storing a large amount of user **interface** representations on a back-end database host, thus alleviating performance and space constraint problems on the client or business process hosts. However, for smaller applications, user **interface** representations may not need to be stored on a separate user **interface** repository 152.

Accordingly, a user **interface** converter 212 can generate a header file 247 instead of database files 246...

...GUI files 248 are used by application developers and maintenance personnel to modify application screens and...

...of the re-engineering system 30. In a preferred embodiment of the present invention, the GUI file 248 are written in Microsoft Visual Basic. The application re-engineering process 30 then uses the GUI file to load screen information in Visual Basic, which can be viewed as the graphical user **interface** editor 310, make any modification in Visual Basic, resulting in a modified GUI file, and then run a Visual Basic to Oracle conversion process as described regarding the graphical user **interface** editor 310 to load the modified GUI file into the user **interface** repository database 152, ready for usage by the application...

...The user **interface** conversion utility 210 calls the user **interface** converter 212 to generate the "target" representation just described. In a preferred embodiment of the present invention, the user **interface** converter 212 is an ANSI C program, which takes a MFS file as an input and generates output files. To perform this function, the user **interface** converter 212 can be structured using conventional compiler technology, including a scanner 241, a parser...

...the statements of the source language. In this context, the delimiter that enables the user **interface** converter 212 to determine when the end of a statement has been reached is defined...

...calling process, eliminating the need to maintain a two-way communication structure with the user **interface** module and the accompanying state information in the business layer. Instead, re-architected batch processes are stand-alone programs constituted by a **wrapper** that provides means to parse the input arguments and call the top-level batch job...

...this top-level batch job requires some form of job scheduling infrastructure. As an example, **legacy** Job Control Language (JCL) can be converted to a scripting language-equivalent such as UNIX...

...commands or library calls that provide functionality that is similar to that of the source **legacy** system. In spite of these differences, batch conversion and processing follows the same fundamental principles

...

...target DDL language for increased maintainability and flexibility, as was the case with the user **interface** and procedural language conversion utility. For illustrative purposes, IMS DL/1 can be considered as...

...a high-level therefore, the custom and re-engineering system 30 includes a graphical user **interface** editor 310, a graphical business process editor 320, a graphical business object editor 330, a...

...the application code to be generated automatically from graphical representations. In particular, the graphical user **interface** editor 310 can be a commercially available user **interface** display platforms or **GUI** builders discussed in the context of the presentation layer 110...

...FIG. 28 is a block diagram of the graphical user **interface** editor 310 of FIG. 1. The graphical user **interface** editor 310 is a typical user **interface** made to create menus and paint screens. As such, the graphical user **interface** editor 310 includes a screen editor 311 to position graphical representations of business objects on

...

...labels, buttons, selection boxes, pull down lists, and similar graphical objects that compose a user **interface**. These graphical representations of business objects can be grouped so that a screen can be...

...be displayed across a number of application screens. The screen editor 311 creates internal user **interface** representations 312 which are processed by a user **interface** code generator 313 into data stored in the user **interface** repository 152...stored in the relational tables of the data layer RDBMS. As such, it is an **interface** that provides graphical access to each application table and permits the application developer or maintainer...

1/3,K/14 (Item 13 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4104970 **IMAGE Available

Derwent Accession: 1998-312857

Utility

E/ **System, method and article of manufacture for communications utilizing calling, plans in a hybrid network**

Inventor: Elliott, Isaac K., Colorado Springs, CO

Krishnaswamy, Sridhar, Cedar Rapids, IA

Assignee: MCI Communications Corporations 02), Washington, DC

MCI Communications Corp (Code: 40955)

Examiner: Chin, Wellington (Art Unit: 273)

Assistant Examiner: Carman, Melissa Kay

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 5867495 | A | 19990202 | US 96758734 | 19961118 |

Fulltext Word Count: 98158

Description of the Invention:

...physical network model provides QOS provisions such as wide range of qualities, adequate QOS for **legacy** applications, congestion management and user-selectable QOS...

...requirement in Service Creation is for the engineers who are producing basic capability objects to **insure** each can be reused in many different services as needed...The dbMon typically presents a passive **interface**; data is fed to it. However monitoring is a hierarchical activity and further analysis and...

...dbMon 2240 supports the following **interface** operations...

...The Operations consoles (Ops) 2244 provide the workstation-**interface** for the personnel monitoring, administering, and otherwise managing the system. The Ops consoles provide access...

1/3,K/15 (Item 14 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4104969 **IMAGE Available

Derwent Accession: 1998-312857

Utility

E/ **System, method and article of manufacture with integrated video conferencing billing in a communication system architecture**

Inventor: Krishnaswamy, Sridhar, Cedar Rapids, IA

Elliott, Isaac K., Colorado Springs, CO

Reynolds, Tim E., Iowa City, IA

Forgy, Glen A., Iowa City, IA

Solbrig, Erin M., Cedar Rapids, IA

Assignee: MCI Communication Corporation 02), Washington, DC

MCI Communications Corp (Code: 40955)

Examiner: Chin, Wellington (Art Unit: 273)

Assistant Examiner: Carman, Melissa Kay

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 5867494 | A | 19990202 | US 96752271 | 19961118 |

Fulltext Word Count: 125798

Description of the Invention:

...physical network model provides QOS provisions such as wide range of qualities, adequate QOS for **legacy** applications, congestion management and userselectable QOS...

...requirement in Service Creation is for the engineers who are producing basic capability objects to **insure** each can be reused in many different services as needed...The dbMon typically presents a passive **interface**; data is fed to it. However monitoring is a hierarchical activity and further analysis and...

...dbMon 2240 supports the following **interface** operations...

...The Operations consoles (Ops) 2244 provide the workstation-**interface** for the personnel monitoring, administering, and otherwise managing the system. The Ops consoles provide access...

1/3,K/16 (Item 15 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4061950 **IMAGE Available

Derwent Accession: 1998-594401

Utility

EXPIRED

E/ Hybrid processor and method for executing incrementally
upgraded software

; COMPUTER APPARATUS

Inventor: Kirsch, Steven A., Agoura, CA

Mellema, Dwight J., Pasadena, CA

Assignee: Raytheon Company 02), Lexington, MA

Raytheon Co (Code: 69864)

Examiner: Donaghue, Larry D. (Art Unit: 273)

Combined Principal Attorneys: Alkov, Leonard A.; Lenzen, Jr., Glenn H.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 5828897 | A | 19981027 | US 96769571 | 19961219 |

Fulltext Word Count: 4699

Description of the Invention:

...shown as a functional block diagram that contains a DoD-STD 1750A processor 11 (or **legacy** processor 11) containing dual central processing units (CPUs) 12a, 12b and dual program memories 13a...

...local bus 18, allowing tightly-coupled functions to operate on the processors 11, 20. An **interface** 15 is provided as part of the DoD-STD 1750A processor 11 that couples the...

...input/output path for the processor 20 via its local bus 18 and an optional **interface** 19...

...in effect, creates a multi-ported memory 14 that is implemented using a relatively inexpensive **interface** circuit 16, such as a field programmable gate array 16, that interfaces the advanced processor...

...processor 11 are not accessing the memory 14, microtiming changes in the execution environment of **legacy** software 23 (FIG. 2) running on the central processing units 12a, 12b of the 1750A...

...processors 11, 20 provides a physical mechanism in which shared variables 24 are accessible to **legacy** and new application software 23, 25 (FIG. 2) executing on both processors 11, 20. However...

...by the software tools which compile and link the application software 23 running on the **legacy** processor 11, but then the application software 25 running on the 32-bit processor 20...

...the operand memory 14 using a native representation of the instruction set architecture of the **legacy** processor 11. However, these shared variables or operands 24 are typically incompatible with the instruction ...

...The **legacy** shared operand 24 implementation is thus hidden from the upgraded or new application software 25. Only knowledge of the **interface** specification of the shared operand 24 is required. As **legacy** software 23 is migrated to the new language, object (operand 24) implementations change, but do not affect the previously migrated code as long as the **interface** specification is maintained ...

...The present software architecture 30, in effect, creates a **wrapper** around the existing **legacy** software 23. The object request broker 21 manages the **wrapper** to insure a consistent object **interface** to the shared memory 14...

...the addition of a symmetrical object request broker (not shown) that is used by the **legacy** software 23. Shared objects (variables or operands 24) may then be stored in the operand...

...object request broker 21 is disposed 43 on the second processor 20 to provide an **interface** between the software 25 that runs on the second processor 20 and the reference depository...

...request broker (not shown) is disposed 47 on the first processor 11 to provide an **interface** between the software 23 that runs on the first processor 11 and the second reference...

1/3,K/17 (Item 1 from file: 761) [Links](#)
Datamonitor Market Res.
(c) 2007 Datamonitor. All rights reserved.
00037547

Global Round-up: 1.32 INDUSTRY COMMENT

Main Title: MarketWatch

Publication Date: December 06, 2005

Source: DATAMONITOR

Telephone: +44 20 7675 7000

Word Count: 4519 (2 pp.)

Language: English

Country: WORLD

Industry: MARKETING SERVICES

Company Names (DIALOG Generated): Actona ; Akimbi Systems ; Another Microsoft ; Apple ; Autonomy 's Intelligent Data ; Autonomy Corporation ; Autonomy Inc ; AutoZone ; Azul Systems ; ACNS ; AT&T ; Bank of America ; Blum Capital Partners ; Borland ; Both IBM ; Brasil Telecom ; Butler Group ; BT Global Services ; BT Infonet ; Cambridge Technology Partners ; Cardiff Software ; Celerant Consulting ; Cesky Telecom ; Chilean ; Cisco Systems ; Clearing Corporation ; Comicro ; Community Technology ; Compute Appliances ; Computer Associates ; Computer Reseller News ; Content Networking Software ; Credit Suisse First Boston ; Dralasoft ; Eircom Group ; Enterprise Resource Planning ; Equally Cisco ; European Commission ; European Software Association ; Expand Networks ; EMC ; EMEA ; FolderShare ; Follow Novell ; Further M & A ; FTC ; General Motors ; Global Address List ; Golden Gate Capital ; Goodyear ; Google ; Grupo Santander ; Hallmark Cards ; Hewlett Packard ; Infor Global ; Informix ; Internet Corp ; IBM Global Services ; ICANN ; ING ; ISP ; IT Management ; Java Virtual Machine ; Legadero Software ; Lifetime Products ; Lilly Software ; Live Safety Center ; Lumin ; Luminating ; Macnica Networks ; Mercury Interactive ; Messaging Management ; Meteor Mobile Communications ; Microsoft 's Media Center ; Microsoft Business Solutions ; Microsoft Corp ; Microsoft Exchange ; Microsoft Office ; Motion ; Motorola ; Nokia ; Novell ; NCR Corporation ; Open Invention Network ; Open Source Development Labs ; Oracle ; Overstock ; Pearl Group ; Philips ; Pixar ; Providence Equity Partners ; RealNetworks ; Red Hat ; Royal Philips Electronics ; Security Center ; Sony ; Sun 's Open Office ; Sun Microsystems ; Sunopsis ; Surgient ; Symantec Corp ; SCO ; Tally Systems Corp ; Tata Consultancy Services ; Teradata ; Time Warner ; Trend Micro ; TCS/Pearl Group ; TFT ; TPI ; Unisys ; US Federal Trade Commission ; Veritas ; Verity ; VeriSign ; Virtual Machine ; Vodafone ; Webroot Software ; Windows Live Safety Center ; WAFS ; WAN ; WDS ; Yahoo

...the operation of life and pensions administration for the closed book portfolio of UK-based insurer Pearl Group. The deal involves the transfer of existing Pearl non-IT staff to TCS...

...950 of Pearl Group's 1,100 staff, as well as leasing buildings at the **insurer's** existing site in Peterborough, UK. TCS plans to establish a business process outsourcing (BPO...

...an overhead in terms of the cost of extra systems and business administration required. Another **insurer**, Swiss Re, set up a subsidiary around eight years ago to buy such policies, and...into one single solution, enabling users to gain connectivity through an easy-to-use client **interface** and, depending on location, to exploit a number of different access technologies, such as Wi...

...Vista CTP that anti-spyware would be bundled is the fact that the Security Center **interface**, introduced in Windows XP Service Pack 2, now contains a reference to spyware alongside anti...as cutting costs by targeting Novell's two corporate jets, its "overstaffed" R&D department, **legacy** products, and its 400 or so NetWare engineers, as well as selling non-core businesses...

...in October.

The question is, where next for cost cutting at Novell? As well as **legacy** sales positions, clearly the R&D department should be a focus, according to Blum Capital...

...web standards that enables features like drag-and-drop components. AJAX enables a slick user **interface** that works more like a desktop app than a typical web site. There are no...

...respects, all of the aforementioned Live services, including Office Live (see separate article), are a **wrapper** around a core search-advertising engine, Live Search and adCenter.

MSN, currently Microsoft's flagship...business and allow for flexible growth opportunities for the mobile unit with no shackles to **legacy** communications. It sold its Eircell operation to mobile giant Vodafone for E4.3 billion in...

1/3,K/18 (Item 1 from file: 991) Links

NewsRoom 2006

(c) 2007 Dialog. All rights reserved.

1189559964 178C1UKV

From promises to prime time: early adopters got into the Web-services game a while ago, but most insurers took a wait-and-see stance and held off for proof of performance. The time may now be ripe to get on board--but jumping on without a good strategy could make for a rough ride.(TECHNOLOGY)

Meyer, David

Risk & Insurance , v 17 , n 5 , p 74

Saturday , April 15, 2006

Journal Code: ARBR **Language:** English **Record Type:** Fulltext

Document Type: Magazine **ISSN:** 1050-9232

Word Count: 1,501

...recently, the majority of Web services have been implemented in their simplest form with some **insurers** using services as a **wrapper** for their **legacy** system to expose the older technology to the Internet--a technique commonly referred to as Web-enabling. While the resulting user **interface** may be based on Web services, the underlying technology remains the **legacy** architecture with all its maintenance and integration challenges.

Web services have also been commonly used...

?

1/3,K/2 (Item 1 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

5882020 **IMAGE Available

Derwent Accession: 2005-009938

Utility

System and method for dynamically adjusting to CPU performance changes

Inventor: Cooper, Barnes, Aloha, OR

Assignee: Intel Corporation 02), Santa Clara, CA

Examiner: Wiley, David (Art Unit: 213)

Assistant Examiner: Avellino, Joseph E.

Law Firm: Schwegman, Lundberg, Woessner & Kluth, P.A.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| Main Patent | US 6823516 | A | 20041123 | US 99371751 | 19990810 |

Fulltext Word Count: 18816

Description of the Invention:

...support mechanism for performance state transitions consists of the following two software elements: a user **interface** and a device driver. In one embodiment, the user **interface** is an extension of the operating system's configuration/power management application (e.g., the...

...In one embodiment a WDM device driver receives mode commands from the user **interface**. In one embodiment, mode settings are stored in the system registry. The driver registers for...

...When using SMIs to implement performance state transitions in **legacy** operating systems, it is recommended that they be set up on all pertinent events (e...as they are received. Once a performance state transition application has been granted the transition **interface**, the application periodically queries the SMI handler for system status. These status calls return information...

...one such embodiment, the application reads the user configuration settings from a control panel user **interface**. By using all of this data, the application can then request that the SMI handler...

...the application. The application can then note the new performance state and update the user **interface**, or retry the operation at a later time if it failed...

...to 170 and SMM support for performance state transitions is disabled, as

is the application **interface**.

...

...moves to 174, where SMM support for performance state transitions is disabled and the application **interface** is enabled...

...disable command. If so, control moves to 180, the application is disabled and the SMI **interface** is enabled...

...SMI handler to cease managing performance state transitions, as well as effectively disabling the SMI **interface** to the performance state transition application. If the application is already loaded, the application receives...

...application loads after the OS has enabled native transition support, the enable performance transition application **interface** will fail the request. In either case, the application will unload once the OS has ...

...routine with the segment registers setup appropriately. In one embodiment this is handled by a **wrapper** routine that performs the segment loading...

...tasks for managing a performance-state-enabled system. The first is that it provides the **interface** to the Performance State Transition Applet for reading status, checking and setting capabilities, and performing...

...The following components are covered in this implementation: the Performance State Transition Applet SMM **Interface** and the State Transition Event Handlers. The Performance State Transition Applet SMM **Interface** routine processes software SMIs from the Performance State Transition Applet, or in response to transition...

...In one embodiment, the Performance State Transition Applet SMM **Interface** is accessed as a hook to the processing of generic software SMIs. In one such...

...values stored in the CPU's general-purpose registers. The next section describes this command **interface** in detail...

...mode, as the OS or applet is in control. When neither the OS nor applet **interface** is in control, the SMM handler can perform performance state transitions in response to AC...

...WINDOWS NT 4.0, WINDOWS 98 and WINDOWS 2000 operating system via an SMM-based **interface**. An applet is provided that communicates to the platform system management interrupt (SMI) handler to...

...one such embodiment, the SMI handler performs all performance state transitions so that it can **insure** that OS registers are properly saved/restored in order to prevent collision with accesses by...

...are applied to operating systems other than WINDOWS 98 and WINDOWS 2000.

This is because **legacy** non-ACPI compliant operating systems used CPU speed dependent calibration loops for driver timing routines...

...3) SMI+Applet (**legacy** non-ACPI compliant operating systems). A one-time transition from high to low frequency is...

...The second SMI command port value is used for the performance state transition applet **interface**. In this case, in order to claim only one additional SMI command port value and...

...To keep the **interface** isolated from collisions from ACPI OS commands, in one embodiment the **interface** uses only one register to initiate the SMIs. In addition, returned data is reported back...

...In one embodiment, the following commands can be used to communicate with the SMM **interface**: Performance State Disable, Performance State Control, Get Performance State Status, Set Performance State, and Set...

...this command has two subfunctions that allow the applet to enable and disable the SMM **interface**. However, the Get Performance State Status function is always enabled...

...which setup mode the user has selected, AC status, and the revision of the SMM **Interface** it is compliant with. (NOTE: This function must maintain memory copies of all variables to...

...user BIOS setup options to reflect changes made by the user in the applet user **interface**. The setup options include disabling performance state transitions, operating in one particular state all of

1/3,K/3 (Item 2 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

5556067 **IMAGE Available

Derwent Accession: 2001-191263

Utility

E/ **Braille computer monitor**

Inventor: Becker, John V., late of New York, NY, deceased

Becker, David A., 87 Remsen St., Brooklyn, NY, 11201, legal representative

Hinton, Daniel E., 815 Galway Garth, Arnold, MD, 21012-1342

Anderson, Jr., Hugh G., 1755 Manchester Rd., Westminster, MD, 21157

Assignee: Unassigned

Unassigned Or Assigned To Individual (Code: 68000)

Examiner: Liang, Regina (Art Unit: 264)

Law Firm: Katten Muchin Zavis Rosenman

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6700553 | A | 20040302 | US 2002131542 | 20020424 |
| Division | US 6417821 | A | | US 2000605514 | 20000628 |

Fulltext Word Count: 5657

Description of the Invention:

...belt power transmission system 800, and a support structure 900. Not shown are a computer **interface**, an outer casing, a computer and an electronic control package of generally known arrangement...

...100 have a spring that returns them to the raised position. The twice tapping requirement **insures** intentional versus accidental communicative desires and is similar to double clicking with a mouse...

...deflections imposed by dot pin drag forces. Part of the difficulty was from working with **legacy** dot pin dimensions. Therefore, in configuring an erasing mechanisms 400 especially for multi-line TCMs...

...The user **interface** for the erasing mechanism 400 is a hand operated electronic switch that is surface mounted...

...A standard computer **interface** is connector-mounted on the traveling printhead 300 and connects the TCM to a standard...

...wired to the connector. The outer casing of the TCM is a conformal shell or **wrapper** that provides a smooth **interface** for the user while preventing the user from exposure to the inner workings of the...

1/3,K/4 (Item 3 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

5364989 **IMAGE Available

Derwent Accession: 2003-067466

Utility

E/ **System and methods using a system-on-a-chip with soft cache**

Inventor: North, Gregory Allen, Austin, TX

Assignee: Cirrus Logic, Inc. 02), Austin, TX

Cirrus Logic Inc (Code: 35367)

Examiner: Bragdon, Reginald G. (Art Unit: 218)

Assistant Examiner: Song, Jasmine

Combined Principal Attorneys: Murphy, Esq., James J.Winstead Sechrest & Minick, P.C.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6622208 | A | 20030916 | US 2001822645 | 20010330 |

Fulltext Word Count: 9219

Description of the Invention:

...ROM) 137 operate from main bus 103 via a local AHB bus 104 and an **interface** 105 which bridges local AHB bus 104 and main AHB bus 103. This configuration minimizes...

...off main AHB bus 103 are a 4-channel DMA engine 106, and flash/SRAM **interface** 107, including an external memory controller, which maps up to 512 MByte external memory into the microprocessor memory space as an extension of on-chip memory, a test **interface** controller (TIC) 108, arbiter 109 and LCD **interface** 110. Test **Interface** Controller (TIC) 108 can take over the bus control from microprocessor 101 and mimic the...

...connected to AHB/APB buses. Arbiter 109 arbitrates bus requests on main bus 103. LCD **interface** 110 supports connections to various LCD panels (since the display may require a large frame...

...An AHB-DSP **interface** 111, which is a slave to main bus 103, allows microprocessor 101 to move data...

...The peripherals operating from APB bus 112 include a USB slave **interface** 114 which supports communications between system 100 and a personal computer (PC) or similar device. When system 100 is used in a portable digital music appliance, this **interface** enables the quick

downloading files from the PC to the portable audio system. UARTa115 is ...

...serial port is fully 16550 compatible and supports various baud rates. It also provides a **legacy** communication channel to an associated PC...

...as Multi-Media Card (MMC). A master mode compatible I2C port 118 provides another common serial **interface** to a range of devices such as EEPROM, DAC/Codecs and some displays...

...GFace 132 interfaces DSP 102 with main bus 102, through slave AHB/DSP **interface** 111, and with the DSP memory. In the illustrated embodiment, DSP 102 is associated with...

...AHB to DSP Slave **interface** 111 allows microprocessor 101 to send read and write requests to the different local memories 133-135 of DSP 102 and global memory 136. Additionally, **interface** 111 synchronizes the microprocessor and DSP clock domains and performs the necessary handshaking. In particular, **interface** 111 responds to transaction requests from the currently granted local or global AHB master 203...

...AHB to DSP slave **interface** 111 operates between two different clock domains. Preferably, the frequency of the DSP clock domain...

...a faster clock. Preferably, a signal is taken from the clock generator which indicates to **Interface** 111 the last DSP cycle before the next microprocessor clock domain rising edge. From this...

...perform byte, half-word and word access to both the RAM and ROM sections. A **wrapper** 138 makes the RAM and ROM AHB compliant slave devices. Since the preferred ROM space is 16-bit wide only, when Microprocessor 101 performs a word-read, **wrapper** 138 issues two consecutive reads to the ROM and concatenates the two read results into ...

...debugging purpose. When TIC mode is enabled, TIC uses 32 pins of the external memory **interface** as a 32-bit bi-directional data bus 301. An external clock (EXTCLKI) is used...

...Display **interface** 110 includes an LCD Display Controller which supports an **interface** to any one of a number of LCD displays. In particular, system 100 can drive...

...FIG. 4 is a functional block diagram of the preferred LCD **interface**/display controller 110. In this embodiment, the display **interface** includes both an AHB bus master 401 and an ...is generated by dividing down the pixel clock as a function of width of the **interface** bus to the external device, and is used internally for such operations as data muxing...

...after receiving an End of Frame signal from Bus Master 401. MCLK is used to **insure** that the display driver voltage frequency does not fall

to DC...

...bus access priority f highest to the lowest as follows: (1) TIC 108; (2) display **interface** 110; (3) DMA controller 106; and (4) Local/Main AHB **Interface** 105...

1/3,K/5 (Item 4 from file: 654) Links
US PAT.FULL.
(c) Format only 2007 Dialog. All rights reserved.

5340267 **IMAGE Available
Derwent Accession: 1999-277009
Utility
REASSIGNED
E/ Integrated customer web station for web based call
management

Inventor: Baker, Thomas E., Monument, CO
Chaffee, Susan L., Colorado Springs, CO
Chen, Yuchien, Colorado Springs, CO
Gruber, Charles J., Monument, CO
Fishman, Howard P., Colorado Springs, CO

Assignee: WorldCom, Inc. 02), Clinton, MS
WorldCom Inc (Code: 48594)

Examiner: Kizou, Hassan (Art Unit: 262)

Assistant Examiner: Pezzlo, John

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6611498 | A | 20030826 | US 98159506 | 19980924 |

Fulltext Word Count: 20410

Description of the Invention:

...3) a data architecture detailing the application, back-end or
legacy data sources available for networkMCI Interact; and...

...pending U.S. patent application U.S. Ser. No. 09/159,695 entitled
INTEGRATED CUSTOMER **INTERFACE** SYSTEM FOR COMMUNICATIONS NETWORK
MANAGEMENT, the disclosure of which is incorporated herein by reference
thereto...

...A third or back-end tier 18 having applications directed to
legacy back-end services including database storage and retrieval
systems and one or more database servers for accessing system resources
from one or more **legacy** hosts...

...in co-pending U.S. patent application Ser. No. 6,115,040, entitled
GRAPHICAL USER **INTERFACE** FOR WEB ENABLED APPLICATIONS, the
disclosure of which is incorporated herein by reference thereto, the
customer workstation includes client software capable of providing a
platform-independent, browser-based, consistent user **interface**
implementing objects programmed to provide a reusable and common
GUI abstraction and problem-domain abstractions. More
specifically, the client-tier software is created and distributed...

...Intranet Dispatcher Server 26; and the MCI Intranet Application servers 30, and the data warehouses, **legacy** systems, etc. 40...

...client applications responsible for presentation and front-end services. Its functions include providing a user **interface** to various MCI services and supporting communications with MCI's Intranet web server cluster 24...

...mentioned, co-pending U.S. patent application Ser. No. 6,115,040 entitled GRAPHICAL USER **INTERFACE** FOR WEB ENABLED APPLICATIONS, the client tier software is responsible for presentation services to the ...The primary common object services include: graphical user **interface** (GUI); communications; printing; user identity, authentication, and entitlements; data import and export; logging and statistics; error...

...after a successful log on. The backplane 12, inter alia, presents a user with an **interface** for networkMCI Interact application management. A typical user display provided by the backplane 12 may...

...may utilize common object services provided by the backplane 12. FIG. 3 shows graphical user **interface** objects 56a,b created and used by a respective application 54a,b for its own...

...FIG. 4 illustrates an example client GUI presented to the client/customer as a browser web page 250 providing, for example, a...

...As shown in FIGS. 3 and 4, the browser resident GUI of the present invention implements a single object, COBackPlane which keeps track of all those...

...to application objects by name. Once retrieved in this manner, the application object's public **interface** may be used directly...

...pass data therebetween is more fully described in the above-referenced, copending application GRAPHICAL USER **INTERFACE** FOR WEB ENABLED APPLICATIONS...

...in co-pending U.S. patent application Ser. No. 09/159,695 entitled SECURE CUSTOMER **INTERFACE** FOR WEB-BASED DATA MANAGEMENT, the contents and disclosure of which are incorporated by reference...

...server. Thus, an application server not only can offer its browser a client to server **interface** through the proxy, but also may offer all its services from its proxy to other...

...include each customer's network management information and data. As shown in FIG. 2, other **legacy** platforms 40(b), 40(c) and 40(d), 610 may communicate individually with the Intranet servers for servicing specific transactions initiated at the client browser. The illustrated **legacy** platforms 40(b)-(d), 610 are illustrative only and it is understood other **legacy** platforms may be interpreted into the network architecture illustrated in FIG. 2 through an intermediate...

- ...All reporting is provided through a Report Requestor **GUI** application **interface** which supports spreadsheet presentation, a variety of graph and chart type presentations, or both simultaneously...
- ...may be maintained to hold the common configuration data which may be used by the **GUI** applications and by the mid-range servers. Such common data includes but are not limited...
- ...maintained by the customer. The call manager webstation 630 includes a web-based graphical user **interface** (**GUI**) application which enables the customers to define their call terminations, and provision routing rules and associated tabular data to control routing by the SCP 610. The **GUI** application also presents alarms and near real time graphical displays of peg counts and ACD...
- ...structures and function profiles may be performed via the call manager webstation's web-based **GUI** application...
- ...Internet connectivity have standard browsers executing Java applets, hereinafter referred to also as a client **GUI** application, downloaded from the web server 632. The web server 632 which is located in...
- ...the network MCI Interact, include Java class files, but no storage of customer data to **insure** data security. Preferably, more than one web server may be provided for redundancy and fail...
- ...As described above, the client webstation 630 provides a web-based graphical user **interface** (**GUI**) offering data management and data presentation features for the call manager system. The web-based front-end **GUI** is typically written using the Java programming language to **insure** platform independence. The client webstation 630 typically includes a web browser with Java applets for the **interface** for providing access to the call manager webstation application from a standard web browser, e...
- ...client webstations) 630 when the Uniform Resource Locator (URL) for the call manager webstation client **GUI** application is accessed...
- ...The call manager webstation client **GUI** application of the system of the present invention is invoked by clicking an icon labeled...
- ...objects shown in FIG. 9. FIG. 9 illustrates the typical objects making up the client **interface** code in one embodiment of the present invention. The user **interface** classes 634 represent the main **GUI** objects for performing call manager specific functionality. Each of the classes, i.e., user and...
- ...FIG. 7 at 640) via the web server 632 are conducted using the common gateway **interface** (CGI). Requests from the client are typically first targeted at a CGI program, which then...
- ...In a preferred embodiment, a Netscape Server Application Program **Interface** (NSAPI) module may be used as an alternative to the CGI

layer, the NSAPI module...described above, the web server 632 provides a communication pass-through between the web client **GUI** application 630 and the back-end call manager integrated data server (CMIDS) 640 which may...

- ...routing engine, e.g., SCP 610. The CMIDS includes databases 642a-c and provides an **interface** to the call manager SCP 610 for rules writing and list management. The CMIDS databases...
- ...as "man machine language" (MML) commands. The CMIDS 640 utilizes MML as well as other **interface** mechanisms supported by the SCP 610. The call manager integrated data server (CMIDS) 640 physically...
- ...670 may reside in the CMIDS, and provide the functionalities described above. The user account **interface** software component 643 generally maintains sessions with the SCPs and provides the functions of the...
- ...handler process generally maintains databases 642a-c and provides reporting facilities. The CMIDS back-end **interface** 712 supports a number of **interface** mechanisms including MML and command line access to the SCP, common alarm and logging services...
- ...Call Manager Client **GUI** Application Implementation...
- ...allows for easy handling of multiple views of a data model. The model is a **wrapper** for an application data object. A controller is a lightweight event handling class, which translates **GUI** events into commands for the application. The view is one particular **GUI** representation of the model. In a MVC typical operation, views register with a model, allowing...
- ...of multiple views when the model changes. Each view has a controller, which handles the **GUI** events, and translates them into command descriptions. The model stores command descriptions, which for example...
- ...webstation application allows authorized customers to manage their ACD data networks via a web-based **interface**. Specifically, customers are enabled to provision hierarchies for their business; control all routing of their...
- ...browser at the webstation 630, deploys a backplane applet via which the call manager client **GUI** applications may be invoked...
- ...conduit through which all other client applications may be deployed, including the call manager webstation **GUI** client application. At step 810, the backplane requests a list of authorized applications from the...
- ...on the call manager icon, triggering the backplane to launch the call manager webstation client **GUI** application...
- ...the back-end returns a list of the available SCP to the front-end web

GUI client application. The proxy generally maintains a "routing engine" list having SCP names and their...

...When the front-end web GUI client application receives the list, a list of routing engine names may be displayed in...For executing the testing process, the debugger/tester uses the MML interface to the routing engine, i.e., the debugger/tester formulates the user actions to one...

...a "Backup" option from the administration button menu and invoke the backup functionality. The client GUI application sends a "RTRV-BK-STATUS" message to check the status of the back-end...

...shown in FIG. 15 opens with a list of retrieved gateway types. Typically the client GUI application sends two messages to retrieve information needed to populate the dialog box 980. A...

...the functional areas. For example, strings such as "OK", "Cancel" which are used throughout the GUI, are typically placed in the global list. The class naming convention is "CMXXXStrings" 1044, where...

1/3,K/6 (Item 5 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4854451 **IMAGE Available

Derwent Accession: 2003-438383

Utility

CERTIFICATE OF CORRECTION

E/ **Service installation on a base function and provision of a pass function with a service-free base function semantic**

Inventor: Hunt, Galen C., Bellevue, WA

Assignee: Microsoft Corporation 02), Redmond, WA

Microsoft Corp (Code: 32791)

Examiner: Morse, Gregory (Art Unit: 212)

Assistant Examiner: Das, Chameli C.

Law Firm: Klarquist Sparkman, LLP

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| Main Patent | US 6546553 | A | 20030408 | US 99349732 | 19990708 |
| CIP | Pending | | | US 98197246 | 19981120 |
| CIP | Pending | | | US 98196836 | 19981120 |
| CIP | Pending | | | US 98196974 | 19981120 |

Fulltext Word Count: 17088

Description of the Invention:

...same call signature, including number of arguments and calling convention. Using the same calling convention **insures** that registers are properly preserved and that the call stack is properly aligned between detour...

...A component instantiation function returns a reference to an **interface** of the instantiated component to the client component that called the function. To profile an application, the COIGN instrumentation wraps the **interface** with an **interface wrapper** before returning it to the client component. Subsequent calls to functions on interfaces exposed by the component are intercepted by the **interface wrapper**. Thus, a layer of profiling instrumentation is inserted to measure parameters of function calls...

...installation according to the present invention include timing tests on software components, redirection services for **legacy** support of software, exception handling, and a test harness for a software system...

...keyboards, mice, displays, registries, etc. from any machine in a network. To provide support for **legacy** applications, COP uses the service installation system of the present invention to intercept all application...

1/3,K/7 (Item 6 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4780991

Derwent Accession: 2003-197057

Utility

EXPIRED

E/ Integration of objects including Java bytecodes with legacy
3270 applications

Inventor: Gray, James Peyton, Chapel Hill, NC
Kaminsky, David Louis, Chapel Hill, NC
Mathewson, II, James Merwin, Chapel Hill, NC
Peters, Marcia Lambert, Raleigh, NC
Telford, Richard Dean, Cary, NC

Assignee: International Business Machines Corporation 02), Armonk, NY
International Business Machines Corp (Code: 42640)

Examiner: Harrell, Robert B. (Art Unit: 212)

Assistant Examiner: Kang, Paul H

Law Firm: Myers Bigel Sibley & Sajovec

Combined Principal Attorneys: Ray-Yarletts, Jeanine S.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6480895 | A | 20021112 | US 9873423 | 19980506 |

Fulltext Word Count: 12681

Summary of the Invention:

...so widely used, that systems using the 3270 datastream architecture are often referred to as "**legacy**" systems...

...The 3270 datastream architecture was designed before graphical user interfaces (**GUI**) became commonplace. Because of the ease of use that a graphical user **interface** may provide, it is desirable to provide graphical user interfaces for 3270 datastream architecture systems. However, due to the **legacy** nature of 3270 datastream architecture systems, it is generally desirable to provide graphical user interfaces without requiring rewriting of **legacy** code or rearchitecting of **legacy** systems...

...One widely used programming language that can provide a rich graphical user **interface** is Java. As is well known to those having skill in the art, Java programs...

...Accordingly, it would be desirable to integrate Java with **legacy** 3270 applications to obtain the advantages of Java as a portable programming language and the...

- ...Java's centralized application distribution paradigm. However, it would be desirable to integrate Java with **legacy** 3270 applications without the need to write new **legacy** applications or to modify existing **legacy** applications...
- ...Many techniques exist for integrating Java with **legacy** 3270 applications. These techniques include screen-scraping, an HTML gateway, an object request broker and...
- ...host data from one or more sources with new program logic and a graphical user **interface**. The client application gains access to the 3270 data through an application programming **interface** on a terminal emulator. This program can read the screen contents, issue keystrokes and perform...
- ...screen, or it may interpret the data and convert it into a new graphical user **interface** representation. It is even possible to read-in the 3270 screen maps residing on the...
- ...used to replace 3270 datastream communications with object-oriented remote procedure calls. Thus, access to **legacy** data and **legacy** applications may be provided by wrapping them in an object **wrapper**. An Object Request Broker is an example of an object **wrapper**. Unfortunately, this approach may require monolithic new software development. It may not lend itself to incremental replacement of **legacy** applications. However, with the large amount of **legacy** application code that is present, it may not be feasible to replace these applications monolithically...
- ...The above survey indicates that although it is generally desirable to integrate Java with **legacy** 3270 applications, there may be shortcomings with conventional techniques for doing so. Accordingly, there continues...
- ...It is another object of the present invention to integrate Java with **legacy** 3270 applications without requiring rewriting of **legacy** applications...
- ...is yet another object of the present invention to allow incremental integration of Java into **legacy** 3270 applications...
- ...may be placed on the same computer (the host or primary logical unit), where the **legacy** applications reside. The benefit of Java's centralized application administration and automatic software distribution may...
- ...Java bytecodes are transported to the client through the same conduit that transports the 3270 **legacy** data. Thus, new infrastructure need not be developed, and existing management projects, procedures and tools...
- ...used. Java bytecodes including Java applets, images and/or audio may

thereby be integrated with **legacy** applications, both temporally and spatially. Incremental replacement of **legacy** applications with new graphical user **interface**-based Java applications may be provided...

- ...photograph next to the entry. A travel expense reimbursement form can include a graphical user **interface** calculator or a hot link to the company's travel policy that is stored on a different server. Thus, a new Java applet can be used to annotate an existing **legacy** application, without the need to change the **legacy** application itself. It can be used to add more user-friendly help information and/or reduce training costs. Individual panels may be updated to a graphical user **interface** form incrementally...
- ...bytecodes so embedded between the primary logical unit and secondary logical unit, it should be **insured** that the secondary logical unit supports Java. An initialization protocol according to the present invention...
- ...with the other transmissions in the session, such as character-formatted data sent by the **legacy** application. The structured fields of image and audio may include a "Display or Play Now..."
- ...than Java bytecodes. Thus, for example, image, audio and other objects may be integrated with **legacy** 3270 applications by embedding the object in a 3270 datastream structured field, and passing the...
- ...Accordingly, Java and other arbitrary objects may be integrated with **legacy** 3270 applications, without requiring the rewriting of **legacy** applications and allowing the use of highly developed 3270 datastream architecture. Initialization protocols, bootstrapping mechanisms...

1/3,K/8 (Item 7 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4755064 **IMAGE Available

Derwent Accession: 2003-074157

Utility

E/ **System and method for managing a plurality of processor performance states**

Inventor: Cooper, Barnes, Aloha, OR

Assignee: Intel Corporation 02), Santa Clara, CA

Intel Corp (Code: 42458)

Examiner: Ray, Gopal C. (Art Unit: 211)

Law Firm: Schwegman, Lundberg, Woessner & Kluth, P.A.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6457135 | A | 20020924 | US 99371268 | 19990810 |

Fulltext Word Count: 18589

Description of the Invention:

...bootstrapped as normal. For the situation where an ACPI-operating system is installed and a **legacy** solution is enabled, in one embodiment the SMI handler immediately ceases to manage performance state ...support mechanism for performance state transitions consists of the following two software elements: a user **interface** and a device driver. In one embodiment, the user **interface** is an extension of the operating system's configuration/power management application (e.g., the...

...In one embodiment a WDM device driver receives mode commands from the user **interface**. In one embodiment, mode settings are stored in the system registry. The driver registers for...

...When using SMIs to implement performance state transitions in **legacy** operating systems, it is recommended that they be set up on all pertinent events (e...

...as they are received. Once a performance state transition application has been granted the transition **interface**, the application periodically queries the SMI handler for system status. These status calls return information...

...one such embodiment, the application reads the user configuration settings from a control panel user **interface**. By using all of this data, the application can then request that the SMI handler...

...the application. The application can then note the new performance state and update the user **interface**, or retry the operation at a later time if it failed...

...to 170 and SMM support for performance state transitions is disabled, as is the application **interface**.

...

...moves to 174, where SMM support for performance state transitions is disabled and the application **interface** is enabled...disable command. If so, control moves to 180, the application is disabled and the SMI **interface** is enabled...

...SMI handler to cease managing performance state transitions, as well as effectively disabling the SMI **interface** to the performance state transition application. If the application is already loaded, the application receives...

...application loads after the OS has enabled native transition support, the enable performance transition application **interface** will fail the request. In either case, the application will unload once the OS has ...

...routine with the segment registers setup appropriately. In one embodiment this is handled by a **wrapper** routine that performs the segment loading...

...tasks for managing a performance-state-enabled system. The first is that it provides the **interface** to the Performance State Transition Applet for reading status, checking and setting capabilities, and performing...

...The following components are covered in this implementation: the Performance State Transition Applet SMM **Interface** and the State Transition Event Handlers. The Performance State Transition Applet SMM **Interface** routine processes software SMIs from the Performance State Transition Applet, or in response to transition...

...In one embodiment, the Performance State Transition Applet SMM **Interface** is accessed as a hook to the processing of generic software SMIs. In one such...

...values stored in the CPU's general-purpose registers. The next section describes this command **interface** in detail...

...mode, as the OS or applet is in control. When neither the OS nor applet **interface** is in control, the SMM handler can perform performance state transitions in response to AC...

...Windows 95, Windows NT 4.0, Windows 98 and Windows 2000 via an SMM-based **interface**. An applet is provided that communicates to the platform system management interrupt (SMI) handler to...

...one such embodiment, the SMI handler performs all performance state

transitions so that it can **insure** that OS registers are properly saved/restored in order to prevent collision with accesses by...

...are applied to operating systems other than Windows 98 and Windows 2000. This is because **legacy** non-ACPI compliant operating systems used CPU speed dependent calibration loops for driver timing routines...

...3) SMI+Applet (**legacy** non-ACPI compliant operating systems). A one-time transition from high to low frequency is...

...The second SMI command port value is used for the performance state transition applet **interface**. In this case, in order to claim only one additional SMI command port value and...

...To keep the **interface** isolated from collisions from ACPI OS commands, in one embodiment the **interface** uses only one register to initiate the SMIs. In addition, returned data is reported back...

...In one embodiment, the following commands can be used to communicate with the SMM **interface**: Performance State Disable, Performance State Control, Get Performance State Status, Set Performance State, and Set...

...this command has two subfunctions that allow the applet to enable and disable the SMM **interface**. However, the Get Performance State Status function is always enabled...

...which setup mode the user has selected, AC status, and the revision of the SMM **Interface** it is compliant with. (NOTE: This function must maintain memory copies of all variables to...

...user BIOS setup options to reflect changes made by the user in the applet user **interface**. The setup options include disabling performance state transitions, operating in one particular state all of ...

1/3,K/9 (Item 8 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4739012 **IMAGE Available

Derwent Accession: 2002-705418

Utility

CERTIFICATE OF CORRECTION

E/ **Thermal control within systems having multiple CPU performance states**

Inventor: Cooper, Barnes, Aloha, OR

Assignee: Intel Corporation 02), Santa Clara, CA

Intel Corp (Code: 42458)

Examiner: Ray, Gopal C. (Art Unit: 211)

Law Firm: Schwegman, Lundberg, Woessner & Kluth, P.A.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6442700 | A | 20020827 | US 99371381 | 19990810 |

Fulltext Word Count: 21249

Description of the Invention:

...bootstrapped as normal. For the situation where an ACPI-operating system is installed and a **legacy** solution is enabled, in one embodiment the SMI handler immediately ceases to manage performance state ...support mechanism for performance state transitions consists of the following two software elements: a user **interface** and a device driver. In one embodiment, the user **interface** is an extension of the operating system's configuration/power management application (e.g., the...

...In one embodiment a WDM device driver receives mode commands from the user **interface**. In one embodiment; mode settings are stored in the system registry. The driver registers for...

...When using SMIs to implement performance state transitions in **legacy** operating systems, it is recommended that they be set up on all pertinent events (e...as they are received. Once a performance state transition application has been granted the transition **interface**, the application periodically queries the SMI handler for system status. These status calls return information...

...one such embodiment, the application reads the user configuration settings from a control panel user **interface**. By using all of this data, the application can then request that the SMI handler...

...the application. The application can then note the new performance state and update the user **interface**, or retry the operation at a later time if it failed...

...to 170 and SMM support for performance state transitions is disabled, as is the application **interface**.

...

...moves to 174, where SMM support for performance state transitions is disabled and the application **interface** is enabled...

...disable command. If so, control moves to 180, the application is disabled and the SMI **interface** is enabled...

...SMI handler to cease managing performance state transitions, as well as effectively disabling the SMI **interface** to the performance state transition application. If the application is already loaded, the application receives...

...application loads after the OS has enabled native transition support, the enable performance transition application **interface** will fail the request. In either case, the application will unload once the OS has ...

...routine with the segment registers setup appropriately. In one embodiment this is handled by a **wrapper** routine that performs the segment loading...

...tasks for managing a performance-state-enabled system. The first is that it provides the **interface** to the Performance State Transition Applet for reading status, checking and setting capabilities, and performing...

...The following components are covered in this implementation: the Performance State Transition Applet SMM **Interface** and the State Transition Event Handlers. The Performance State Transition Applet SMM **Interface** routine processes software SMIs from the Performance State Transition Applet, or in response to transition...

...In one embodiment, the Performance State Transition Applet SMM **Interface** is accessed as a hook to the processing of generic software SMIs. In one such...

...values stored in the CPU's general-purpose registers. The next section describes this command **interface** in detail...

...mode, as the OS or applet is in control. When neither the OS nor applet **interface** is in control, the SMM handler can perform performance state transitions in response to AC...

...Windows 95, Windows NT 4.0, Windows 98 and Windows 2000 via an SMM-based **interface**. An applet is provided that communicates to the platform system management interrupt (SMI) handler to...

- ...one such embodiment, the SMI handler performs all performance state transitions so that it can **insure** that OS registers are properly saved/restored in order to prevent collision with accesses by...
- ...are applied to operating systems other than Windows 98 and Windows 2000. This is because **legacy** non-ACPI compliant operating systems used CPU speed dependent calibration loops for driver timing routines...
- ...3) SMI+Applet (**legacy** non-ACPI compliant operating systems). A one-time transition from high to low frequency is...
- ...The second SMI command port value is used for the performance state transition applet **interface**. In this case, in order to claim only one additional SMI command port value and...
- ...To keep the **interface** isolated from collisions from ACPI OS commands, in one embodiment the **interface** uses only one register to initiate the SMIs. In addition, returned data is reported back...
- ...In one embodiment, the following commands can be used to communicate with the SMM **interface**: Performance State Disable, Performance State Control, Get Performance State Status, Set Performance State, and Set...
- ...this command has two subfunctions that allow the applet to enable and disable the SMM **interface**. However, the Get Performance State Status function is always enabled...
- ...which setup mode the user has selected, AC status, and the revision of the SMM **Interface** it is compliant with. (NOTE: This function must maintain memory copies of all variables to...
- ...user BIOS setup options to reflect changes made by the user in the applet user **interface**. The setup options include disabling performance state transitions, operating in one particular state all of

1/3,K/10 (Item 9 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4712115 **IMAGE Available

Derwent Accession: 2001-191263

Utility

EXPIRED

E/ **Braille computer monitor**

Inventor: Becker, John V., late of New York, NY, deceased

Becker, David A., 87 Remsen St., Brooklyn, NY, 11201, legal
representative

Hinton, Daniel E., 815 Galway Garth, Arnold, MD, 21012-1342

Anderson, Jr., Hugh G., 1755 Manchester Rd., Westminster, MD,
21157

Assignee: Unassigned

Unassigned Or Assigned To Individual (Code: 68000)

Examiner: Liang, Regina (Art Unit: 264)

Law Firm: Katten Muchin Zavis Roseman

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 6417821 | A | 20020709 | US 2000605514 | 20000628 |

Fulltext Word Count: 5833

Description of the Invention:

...belt power transmission system 800, and a support structure 900. Not shown are a computer **interface**, an outer casing, a computer and an electronic control package of generally known arrangement...

...100 have a spring that returns them to the raised position. The twice tapping requirement **insures** intentional versus accidental communicative desires and is similar to double clicking with a mouse...

...deflections imposed by dot pin drag forces: Part of the difficulty was from working with **legacy** dot pin dimensions. Therefore, in configuring an erasing mechanisms 400 especially for multi-line TCMs...

...The user **interface** for the erasing mechanism 400 is a hand operated electronic switch that is surface mounted...

...A standard computer **interface** is connector-mounted on the traveling printhead 300 and connects the TCM to a standard...

...wired to the connector. The outer casing of the TCM is a conformal shell or **wrapper** that provides a smooth **interface** for the user while preventing the user from exposure to the inner workings of the...

1/3,K/11 (Item 10 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4652243 **IMAGE Available

Derwent Accession: 2000-064334

Utility

E/ **Method for computer internet remote management of a telecommunication network element**

Inventor: Barker, William E., Geneva, IL
Connelly, Lisa M., Lisle, IL
Eggert, Marvin A., Aurora, IL
Foley, Michael P., Elmwood Park, IL
Macfarlane, Kenneth R., Wheaton, IL
Parsons, Philip M., Lisle, IL
Rai, Girish, Bartlett, IL
Rog, Jerome E., Bartlett, IL
Vangsness, Kurt A., Aurora, IL

Assignee: Lucent Technologies, Inc. 02), Murray Hill, NJ
Lucent Technologies Inc (Code: 39644)

Examiner: Rinehart, Mark H. (Art Unit: 212)

Assistant Examiner: Kang, Paul

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| Main Patent | US 6363421 | A | 20020326 | US 9888463 | 19980531 |

Fulltext Word Count: 21821

Description of the Invention:

...The client executes the Client **Interface** and propriety applications via Web pages. Microsoft Internet Explorer and netscape browsers are supported as...

...the web-enabling devices for PCs and X-terminals. Through a Web-based graphical client **interface**, clients' commands generate HTTP requests to the element management system server. The server gathers information...

...network element detailed status display. Client applications communicate with the server via an object oriented **interface** to the element manager API (EMAPI) 55 through the distributed object request architecture (CORBA) 48. This **interface** provides a consistent **interface** to all managed objects in the network, and hides the implementation details associated with the...

...The element management system client 28 is the client's **interface** to the element management system server 32. It consists of the web

browser 45 and...

- ...Web Browser: The web browser 45 is the **interface** to end client, a host for JAVA applets 44, and a virtual machine for JAVA...
- ...Command Line Interpreter 76: provides an ASCII command language **interface** to allow the technician to enter commands at the element management system and observe results...
- ...UX Proxy 78: UX message **interface** (bridge between System V message queues and sockets) to an internal database subsystem (IDS) 79...
- ...Command Handler 88: Handles AP administration command requests issued from either the **GUI** based or text based client interfaces. Executes a RAP 90 to complete the administration request...
- ...Text Command Interpreter 92: A text based **interface** for special situations in which the **GUI** based **interface** presented by the element management system Server is not available to the client...
- ...RAPs 90: The Resource Administration Process is an application processing **interface**(API) for fork-exec'ing a process and obtaining the results of the process execution...
- ...command and control and fault management. These web browser hosted applications provide a graphical client **interface** based client **interface** in a cross ...The client **interface** to the server is described in the EMAPI 55 described in a pending patent application...
- ...by reference herein. The EMAPI 55 is implemented utilizing an industry standard object management group **interface** description language (IDL). The interfaces and semantics of the EMAPI 55 enable client application processes to utilize this **interface** to provide management of the system. Distribution of this **interface** is achieved through use of the Common Object Request Broker Architecture (CORBA) which provides a...
- ...The development of client applications depends only on the EMAPI 55 **interface** specification. The use of CORBA allows the clients to be distributed and implemented in a...
- ...any other system aspects that are not part of an object's **interface**
...
- ...ovspmd (OpenView System Process Management Daemon): This process must be running to enable client-**interface** status-checking programs such as ovstart to work. Since this monitor does not support restarting...
- ...SNMP Collection Daemon which allows clients to define, via the HP OpenView Windows X-based **GUI interface**, SNMP MIB values that are to be collected periodically. It provides ways to define thresholds...

- ...ovw (OpenView Windows): The OpenView Windows X-based **GUI** provides access to map applications, an event browser, and a MIB browser...
- ...The HPOV SNMP API, a C-language **interface** to this runtime system, is provided as part of the HPOVNNM Developer's Kit and...
- ...Provides UX messaging **interface** to the IDS 79 to receive and process IDS triggers. A trigger is needed to...
- ...single message type for forwarding all element management system-generated TI/OP messages. The current **interface** from a mobile switch center (MSC) ECP ROP supports specifying Alarm Level (MAN, INFO, CRIT...
- ...The client **interface** to the services and the managed object attributes and methods is described in the EMAPI...
- ...know whether the underlying protocol to the network element is SNMP, CMIP or a proprietary **interface**). Managed object specific logic is encapsulated within the managed object instead of scattered throughout various...
- ...The definition of these managed object class identifiers and attribute codes is part of the **interface** definition between all service objects and their clients...
- ...interfaces on the AP. They both provide common Ethernet functionality but there is an ethernet **interface** node (EIN) and a LAN managed object...
- ...especially when there are multiple interfaces to it within the server) by placing it within **wrapper** libraries. For example, the Element management system Logger class will provide object-oriented wrappers for ...
- ...**Interface** to clients for starting a session...
- ...**Interface** to clients for manually ending a session...
- ...**Interface** to clients for periodic check-in (heartbeat...
- ...**Interface** to other server components for registering interest in notifications of session/application termination. The components...
- ...Notification to registered entities when a session/application has been ended via the callback notification **interface** described above...
- ...be generated by element management system components on the same machine, or even by other **legacy** network elements. The Event Distributor may be implemented as a set of objects within the...
- ...for the purpose of simple, open-ended event correlation. The Event Screener supports the same **interface** (although not available to clients) as the Event Distributor, but is only for use within...

- ...Provide an IDL **interface** (Event Distributor Only) for registering filters based on the following...
- ...Provide an IDL **interface** for clients (Event Distributor only) and other Object Server components to explicitly cancel a specific...Object Server **Interface**
- ...
- ...Provide an **interface** to managed object classes in the object server to support...
- ...SNMP **Interface**
- ...
- ...The SNMP Mediator handles all interfacing with SNMP agents on network elements. The SNMP **interface** consists of Attribute Polling, Configuration Auditing, Command Execution, SNMP Retry Mechanism, and Trap Delivery...
- ...of sequence, a command request/response convention between manager and agent will be utilized to **insure** that an agent will respond only once to a single command (i.e. SET operation...
- ...be consistent for all GET, GET-BULK and SET operations and will conform to the **interface** prescribed by the HPOVNNM SNMP library or the CMU SNMP library. For further information, see...
- ...be identified through a class dictionary containing attribute codes and type information, available via common **interface** definition to both server and client code. For example, the AP object may contain attributes...
- ...within the element management system server and not by a network element through the SNMP **interface**. The element management system server must generate the command acknowledgment and command response events itself...
- ...Provides **interface** to clients for attribute registration given the following parameters...
- ...Provides **interface** with client for de-registration...
- ...Provides **interface** with Client Session Manager for de-registration when abnormal client termination is detected via audit...
- ...Client **Interface** Components...
- ...components present on a client of the element management system server to provide the client **interface** to NE management. Specific client **interface** style and content will be addressed after the architecture with input from human factors and element management system engineering. In addition to making the client **interface** as easy to use as possible, the client **interface** must retain similarities

with the current maintenance model such that little retraining of the end
...

...The primary client **interface** is provided by an HTML web browser.
Both Netscape navigator and Microsoft Internet Explorer are...
...Java-based **GUI** Infrastructure...

...This section describes base components that are necessary for
implementing the AP specific **GUI** applets. A number of these
components (especially the **GUI** components) may be satisfied (or
based upon) commercial 3rd party products (for example Rogue Wave
JWidgets, or Microline's Grid widget). Also, 3rd party non
GUI container and algorithm classes (either Rogue Wave or JGL for
example) should be considered to...context sensitive command execution
through the use of pull down (or pop up) menus. The **interface** for
command execution and display of command results is the same as the
interface described in the "Command Handler" section above...

...This application provides an alarm browser

interface to active

alarms within the system. The client can specify a filter to limit the...

...Each managed object class will adhere to the **interface** specified
by the managed object base class (provide for client attribute
registration, notification, configuration registration...

...the element management system server is back up. The definitive
requirements for element management system **GUI** Client version
management will be described in the element management system **GUI**
Client Capability Requirements/High-Level Design document...

...MIB is that it contains a very detailed description of the element
management system/AP **interface**. The MIB is intended to serve as
much as possible as a single element management system/AP
interface definition. As such, its details may need to be modified
more frequently than at each...

...with global access permissions for use by infrastructure components
which make use of the same **interface** definition...

...SNMP Agent: Provides the **interface** to the element management
system Server using the SNMP protocol and a MIB defined specifically...

...NE Status API: an **interface** for writing to and reading from the
Network Element Status Table...

...Command/Response API: **interface** between a command source and the
Command Handler for the purpose of issuing commands and...

...ECP Agent: **interface** to Status Display on the ECP;
interface to IDS-AP for processing triggers indicating change in
data. Passes triggers on to interested...

- ...interfaces to the AP via the SNMP Agent. A MIB is used to define the **interface** between the element management system Server and the Agent and is common to both the...
- ...is used by the Manager to determine whether this is an MIB (supporting the other **interface** conventions described in this section), and also to provide a versioning mechanism to support MIB...server infrastructure required to support object services. The following is the Element Management Application Programming **Interface** (EMAPI) 55 in accordance with the invention utilized by EM Clients...
- ...request brokers (ORBs). Application services are provided through object interfaces formally defined in the CORBA **Interface** Definition Language (IDL...
- ...a session. Application identifiers are assigned by the Client. For the Element Manager Graphical User **Interface**, each "window" will be assigned a unique application id. Note that each Client is required...
- ...Each managed object service class must implement the MO **interface**, which defines the following configuration and status services...
- ...client may use this method to register for a snapshot of current status information. This **interface** differs from the previous one in that the requested attribute list may specify any managed...
- ...Each network-element level managed object must also implement the NEMO **interface** which defines additional network-element level configuration services...

1/3,K/12 (Item 11 from file: 654) Links
US PAT.FULL.
(c) Format only 2007 Dialog. All rights reserved.

4639510 **IMAGE Available

Derwent Accession: 2002-205301

Utility

E/ Computer software for converting a general purpose computer network into an interactive communications system

Inventor: Simonoff, Adam J., Fairfax Station, VA

Assignee: The United States of America as represented by the Secretary of the Navy 06), Washington, DC

U S of America Navy Secretary of (Code: 86584)

Examiner: Lim, Krisna (Art Unit: 213)

Combined Principal Attorneys: Bechtel, Esq., James B.; Powell, Jr., Esq., Raymond

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| Main Patent | US 6351777 | A | 20020226 | US 99296746 | 19990423 |

Fulltext Word Count: 16270

Description of the Invention:

...storage device such as a hard disk and a communications device, e.g., a network **interface** card. It should also be mentioned that computers 300a-300r can include desktop computers, laptop...

...that the present invention was developed in response to perceived problems in the interoperability of **legacy** computer hardware used in combat systems and networks and solved those problems. However, since the...

...of the windows for the client-side White Board display, i.e., the White Board **GUI** will be presented to the user. The user can then run the White Board application...

...the White Board displayed on the computers 300a and 300b employ an intuitive graphical user **interface** (**GUI**). Moreover, each user is provided with information regarding his respective White Board. For example, White...selection of an object from the resource list depicted in FIG. 7 creates an empty **wrapper**, which **wrapper** is assigned a unique identifier and which **wrapper** contains the selected object label. Thus, when a mouse down event occurs at step S21

...S24, the information needed to regenerate the selected object is placed into the above-mentioned **wrapper**, to thereby generate a

wrapper object. As mentioned above, the **wrapper** includes a unique identifier so that the **wrapper** object can be locally identified, used by the local White Board client 301 and globally identified to prevent collisions with other **wrapper** objects. During step S26, the **wrapper** object is added to a vector holding all **wrapper** objects drawn on the local White Board client 301. Using the thus generated vector, the **wrapper** object is displayed in the White Board client 301. When a "mouse up" event occurs, the **wrapper** object is transmitted to the White Board server 102 over LAN 400 for relay to...

...up" should be understood to equate to "hard.return" with respect to text, since the **wrapper** object containing text is transmitted to White Board server 102 when a hard return is...

...i.e., both White Board client and White Board server, keeps track of all the **wrapper** objects to be displayed in their respective order via a vector. The vector advantageously stores...

...each object, i.e., each object in the hash table is referred to via a **wrapper** object. Thus, the **wrapper** object tells the White Board the kind of object to display, its location, size and...

...The **wrapper** object provides for an open architecture design so that developing new objects for use with the White Board is greatly simplified. Stated another way, the **wrapper** allows third party objects to simply plug-in to the White Board. The **wrapper** advantageously can be used to define additional characteristics of an object to be displayed without...

...a part of the Java runtime environment. In contrast, optional signifies that a White Board **interface** class may be implemented which requires certain subroutines or methods be available in the source...

...White Board system was developed to permit the White Board server to filter the data, **wrapper** objects, by privilege. Additionally, in order to **insure** traceability, i.e., the ability to retrace or recreate the steps by which the White...

...recreate the same object remotely. In other words, the White Board server time stamps each **wrapper** object so that the White Board system can afterwards determine when the **wrapper** object was created and when the **wrapper** object was modified, and stores a copy of the **wrapper** object on the White Board server (or at a White Board server specified location). Given...

...each user's privilege before deciding whether or not that particular user will receive a **wrapper** object. A more efficient method of relaying **wrapper** objects advantageously can be implemented when all users are equal in privilege. In that case...

...the White Board contents can be changed simultaneously by multiple users with one exception. The **wrapper** around each object allows the White Board system to establish read/write privileges on objects...

- ...As discussed above, the global key name associated with each **wrapper** object contains information regarding the user and White Board client that created that **wrapper** object; the object's **wrapper** also contains the name of the last user to modify the object. Moreover, the White Board server time stamps every change, i.e., every **wrapper** object, as it arrives from one of the White Board clients or as it is...
- ...Moreover, as previously discussed, each **wrapper** object, e.g., each object generated by White Board client 301, derives its unique identifier...
- ...A second Hash Table (named HashTable) contains the object's **Wrapper** associated with a global key name known by ALL privileged White Board clients on the...

1/3,K/13 (Item 12 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4206909 **IMAGE Available

Derwent Accession: 1997-559168

Utility

E/ **System to transition an enterprise to a distributed infrastructure**

Inventor: Eager, Timothy, Fullerton, CA
Anand, Madhav, Cambridge, MA
Aslanian, Edouard, Hermosa Beach, CA

Assignee: i-CUBE 02), Cambridge, MA
i CUBE

Examiner: Hafiz, Tariq R. (Art Unit: 272)

Assistant Examiner: Dam, Tuan Q.

Law Firm: Hamilton, Brook, Smith & Reynolds, P.C.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| Main Patent | US 5960200 | A | 19990928 | US 96714205 | 19960916 |
| Provisional | | | | US 60-16330 | 19960503 |

Fulltext Word Count: 19684

Description of the Invention:

...inter-connected through communication links 170. The data storage layer 150 preferably includes a user **interface** repository 152, a business process repository 154, a business object repository 156, and a data...

...A preferred re-architecting system 20 includes a user **interface** conversion utility 210, a procedural language conversion utility 220, and a data definition language conversion...

...130 and the data access layer 140 of the multi-tier architecture 10. The user **interface** conversion utility 210 is in communication with the user **interface** repository 152 and the data definition language conversion utility 230 is in communication with the...

...A preferred re-engineering system includes a graphical user **interface** editor 310, a graphical business process editor 320, a graphical business object editor 330, a...

...communication with the functionality layer 130 of the multi-tier architecture 10. The graphical user **interface** editor 310, the graphical business process editor 320, the graphical business object

editor 330, and the graphical data record editor 340 are in communication with the user **interface** repository 152, the business process repository 154, the business object repository 156, and the data...

...the architecture 10 supports both custom-developed applications as well as application converted from a **legacy** system. The IMS/VS **legacy** environment is used herein to illustrate, but not limit, architectural concepts that pertain to a converted **legacy** application...

...2 is a functional block diagram of the interrelationships of FIG. 1. Conceptually, the user **interface** translator 210 and the graphic user **interface** editor 310 affect the presentation layer 110 of the multi-tiered architecture 10. The graphical...42 through to the operation stage 48, the amount of operations support required by the **legacy** system decreases and the amount of operations support for the open system increases, as illustrated...

...and performance issues that arise when attempting substantial changes to an application designed for a **legacy** system and converted to a multi-tiered client/server architecture. Custom development refers to the ...

...of support staff for the maintenance phase. One powerful example of integration at the user **interface** layer using the OSI process 469 is the creation of a corporate intranet using internet...

...such as Java from Sun Microsystems to provide a user-friendly, platform independent, common user **interface** to corporate application...

...also take the form of an X-terminal, a workstation console, or a Macintosh style **interface** display. As shown, the presentation layer 110 includes a processor 111 having the current screen...

...also includes internal or external storage, such as a disk device, from which a user **interface** engine is loaded into the memory of the processor 111 as required. For a personal...

...As shown in FIG. 4, the presentation layer 110 includes a user **interface** display platform 115, an application user **interface** representation mechanism 116, and a user **interface** engine 117. In a preferred embodiment of the present invention, the user **interface** display platform 115 is a conventional Graphical User **Interface** (GUI) tool, commercially available. Consequently, the user **interface** display platform 115 has its own internal user **interface** representation mechanism 118 to display the various components of a user **interface**, usually in a graphical way...

...Preferably, the underlying internal user **interface** of the user **interface** display platforms 115 is preferably derived from a frame-based system. A frame system is...

...Similarly to the display platform user **interface** representation

structures 118, the application user **interface** representation structures 116 store descriptive information representative of the different objects that compose a user **interface**. Each object is described by a structure comprising a plurality of fields containing information representing...

...attribute of that object or a relationship between the object and another object. The user **interface** engine 117 maps each of the different objects that compose the user **interface** of a given application into the corresponding representations 118 in the user **interface** display platform 115 of choice for that application...

...On the one hand, the user **interface** engine 117 requests application user **interface** representation structures 116 from the business process layer 120. Once the business process layer 120 satisfies the request, the user **interface** engine 117 converts the application user **interface** representation structures 116 just received into user **interface** representation structures 118 that are expected by the user **interface** display platform 115 for display to the end user on a display station 111...

...through the display station 111, such as selecting an item or modifying information, the user **interface** engine 117 translates that user request from user **interface** display platform representation structures 118 into the corresponding application user **interface** representation structures 116, which are then handed to the business process layer 120 for execution...

...FIG. 5 is a schematic diagram of a sample mapping between application user **interface** representation structures 116 and display platform user **interface** representation structures 118. In the figure, the user **interface** display platform 115 is exemplified as Microsoft Windows 3.x and the display platform user **interface** representation structures 117 are thus the internal Windows 3.x management structures. However, other user **interface** display platforms 115 using similar internal structures to manage windows are supported by the exact same user **interface** engine 117. Notably, the internet's world-wide web, based on the HTML or Java user **interface** languages, is another example of user **interface** display platform 115. Indeed, in a preferred embodiment of the present invention, the user **interface** engine 117 is written using Microsoft Visual C++ and based on the industry-standard Microsoft...

...platform development for Windows 3.x, Windows 95, Windows NT, MacOS, and UNIX-based user **interface** display platforms 115, including internet web servers...

...FIG. 6 is a block diagram of the operational modules of the user **interface** engine 117 of FIG. 4. The user **interface** engine 117 includes an initialization module 117-1, a user input module 117-2, and...

...During initialization, the user **interface** engine 117 first initializes its initial state, setting up any structures necessary for operation. Depending on the implementation, the user **interface** engine 117 can then initialize communications with the business process layer 120, receiving a client identification number. Depending on the implementation, the user **interface** engine 117 can also display an initial application menu or screen, initial objects that are...

...After completing the initialization, the user **interface** engine 117 continues to the user input module 117-2. The user **interface** engine 117 waits for user input and processes it accordingly. In particular, the user input module 117-2 handles interactions with **GUI** objects and performs application-dependent actions in response to user inputs...

...router communications module 117-3. In the state router communications module 117-3, the user **interface** engine 117 creates outgoing application user **interface** representation structures 116 from the screen data and packs these structures for delivery to the business process layer 120. Typically, the outgoing application user **interface** representation structures 116 contain values of screen fields which have changed since the previous call to the business process layer 120. The packed application user **interface** representation structures 116 are then sent to the business process layer 120, which returns packed application user **interface** representation structure 116 describing the result of the transaction. The packed application user **interface** representation structures 116 returned from the business process layer 120 are then unpacked and processed...

...long as the business process layer 120 does not indicate a fatal error, the user **interface** engine 117 processing continues (resumes the wait for user input) at the user input module...

...Most of the user **interface** engine 117 processing occurs in the handling of screens: building a screen from a description...

...more specifically its main state router component (described below), is always initiated by the user **interface** engine 117 because a remote procedure call (RPC) mechanism which interfaces the user **interface** engine 117 with the business process layer 120 is preferably unidirectional and synchronous...

...To simulate asynchronous communication using a unidirectional synchronous RPC model, the user

interface engine 117 includes an ability to periodically poll the state router for messages during the user **interface** engine's 117 idle time, namely when there is no user input to be processed...

...Essentially, during idle message polling the user **interface** engine 117 queries the state router for any initial messages. At the start of an...

...logon screen, application menu, and other object for the user to act upon, the user **interface** engine 117 waits to process user inputs. If the user takes no action and idle message polling is enabled, the user **interface** engine 117 will periodically query the state router for any messages. If message polling is...

...structure, which is preferably a two-way associative array, it is possible for the user **interface** engine 117 to allow window control handlers of the user **interface** display platform 111 to manage general window operation and make callbacks to the user **interface** engine handlers when an action is required, for example, when a button is pressed...

...In a preferred embodiment of the present invention, the user **interface** engine 117 can process any type of action from any type of screen object, e...

...pressed. Typically, when an action is performed, one of two things may happen: the user **interface** engine 117 performs some internal function based on the action, or sends information to be...

...access server runtime environment resides, and a terminal console 136 which serves as a human **interface** for host administration purposes. In addition, a communications controller 138 such as a LAN controller, modem or similar device serves as an **interface** to a communication link. The host computer system 132 can be considered conventional in design...

...is a state router 122. Conceptually, the state router 122 receives requests from the user **interface** engine 117 (FIG. 4) and, based on the request, determines which actions to take. The...

...the state router 122 accepts any resulting return information and forwards it to the user **interface** engine 117...

...The requests received from the user **interface** engine 117 include application user **interface** representation structures 121. The application user **interface** representation structures 121 include request identifiers, transaction codes, screen information, and input/output buffers. A...

...be executed in response to the request. There is one request identifier for any user **interface** event caused by the user. In this regard, request functions are similar to the conventional callbacks found in GUI languages such as X-Windows developed at the Massachusetts Institute of Technology, in Cambridge, Mass...

...FIG. 9 is a flow diagram of the communication mechanism between the user **interface** engine 117 and the state router 122. As depicted, the user **interface** engine 117 includes a user **interface** routine 117-6 and initiates the communication by calling a pass message function 117-8. The pass message function 117-8 first compresses the application

user **interface** representation structures 116 to be transmitted into a single request string using a packing procedure...

...request string compression performed by the packing procedure is necessary because the outgoing application user **interface** representation structures 116 cannot be transferred efficiently as such across the communication link...

...routine 117-9 takes two parameters: the request string to be passed from the user **interface** engine 117 to the state router 122 and the return string to be returned to the user **interface** engine 117 from the state router 122. From the point of view of the state...

...The unpacking procedure converts the request string into an array of request application user **interface** representation structures 121. This array is then passed to a main state router 122-1...

...Once the state router 122 completes its processing, the resulting array of return application user **interface** representation structures 121 is again packed into a return string, which is passed back to the user **interface** engine 117 using an RPC mechanism 122-9...

...Because new application user **interface** representation structures 121 can be added to facilitate the transport of new types of objects... (bit integers), and strings (both variable- and fixed-length). To create a new application user **interface** representation structure 121, a developer need only create packing and unpacking routines for that structure...

...under ANSI C) or using object classes (under ANSI C++). Although the ANSI C language **interface** is very usable, the ANSI C++ language **interface** makes use of object-oriented features such as virtual functions to make packing and unpacking...

...level packing and unpacking routines take arrays (or, in ANSI C++, containers) of application user **interface** representation structures 121 and create a single character string containing the packed information suitable for...

...contains type information as well as member data, so that any sequence of application user **interface** representation structures 121 can be sent and properly reconstructed at the receiving end...

...router 122 of FIG. 8. Initially, when a logon request is received from the user **interface** engine 117 through the request user **interface** structure 121a and then authorized, the state router's 122 internal state is initialized with...loaded from the database repository 152. This screen information is passed back to the user **interface** engine 117 through the return application user **interface** representation structures 121. The user **interface** engine 117 then displays the screen and awaits user input. When a users enters or changes data on a screen and presses a function key, the user **interface** engine 117 translates this user input into a request to

the state router 122...

...and the current state is then updated with any new field values from the user **interface** engine 117. State information is represented by a field state structure 122b. Then, the state...

...which describes a function to be executed, is included with the request from the user **interface** engine 117. The state router 122 verifies the user's authorization to perform this function...

...from the database repository 152, and its information is included in the return application user **interface** representation structures 121b destined for the user **interface** engine 117. The state router 122 then recalls the functionality server that corresponds to the...

...its value. The new values and attributes are then included in the return application user **interface** representation structures 121 array passed back to the RPC mechanism for return to the user **interface** engine 117...

...of custom functionality servers, the state router 122 still processes requests received from the user **interface** engine 117 in response to user **interface** event caused by the user in manner similar to that described earlier. However, much of...

...redirects processing to appropriate functionality servers based on the transaction codes received from the user **interface** engine 117. However, when the functionality server returns, it passes back an event to the...

...data server exists for each of the four application object repositories. Consequently, there is user **interface** data server 141 to manipulate user **interface** objects 142, a business process data server 143 for business processes 144, a business object...

...a database server 147 for application data records 148. The data servers constitute the sole **interface** between the data storage layer 140 and the functionality layer 130, and each data server...

...multiple clients. The set of functions, or services, provided by a server constitutes the server **interface**. This **interface** is specified in an **Interface** Definition Language (IDL) file. The concept of servers is well known, and details of server...

...the present invention, this query string is forwarded to the database using the Oracle Call **Interface** (OCI) from Oracle Corporation. At a high-level, the process consists in initializing bind and...

...150 is a repository for data accessed by the data access layer 140. The user **interface** data repository 152 provides user **interface** objects 153 to the data access layer 140. The business process data repository 154 provides...

...which all database data resides, and a terminal console 155 which serves

as a human **interface** for host administration purposes. DBMS log files are stored in storage unit. A printer 157...

...back, with appropriate error messages posted. Transaction management 161 is useful in distributed systems to **insure** data consistency in the absence of user-defined integrity constraints...

...Network management 166 provides a graphical **interface** to monitor clients, servers, and brokers. Network traffic and performance can thus be monitored, and...

...application version management functions are provided. In addition, currency is handled through locking functions to **insure** data consistency. Data integrity is controllable at the functionality layer by the business objects rules...Returning to FIG. 1, the re-architecting system 20 includes a user **interface** conversion utility 210, a procedural language conversion utility 220, and a data definition language conversion...

...FIG. 16 is a block diagram of the user **interface** conversion utility 210 of FIG. 1. The user **interface** conversion utility 210 converts the user **interface** of an existing application represented by the source user **interface** definitions 211 into target user **interface** definitions 213 using the user **interface** converter 212. In a preferred embodiment of the present invention, the source user **interface** definitions 211 can be viewed as IMS/VS Message Format Service (MFS) files...

...Target user **interface** definitions 213 can take one of three forms: database files 246, a header file 247, and a

GUI file 248.

Database files 246 contain the set of statements necessary to populate user **interface** repository 152 with screen and message information for MFS file 211. In a preferred embodiment...

...A deletion script removes from the user **interface** repository 152 any definitions for the MFS file 211. Once this repository cleanup is accomplished, an insertion script adds to the user **interface** repository 152 any new definitions for the MFS file 211. Consequently, the user **interface** conversion utility 210 can be run multiple times for the same MFS file without negative effects. In a preferred embodiment of the present invention, the user **interface** repository 152 is a standard RDBMS such as Oracle Server 7 from Oracle Corporation

...

...Information stored in the user **interface** repository 152 is converted at application runtime into the user **interface** representation structures of the presentation layer 116. The user **interface** engine 117 of the presentation layer 110 then maps application user **interface** representation structures 116 into display platform user **interface** representation structures 118, used by the user **interface** display platform 115 for display to the user. Accordingly, target user **interface** definitions 213

effectively constitute an intermediary user **interface** definition language for storage of user **interface** information in the user **interface** repository 152 and eventual user **interface** representation structures 118...

- ...an alternative to database files 246. In a preferred embodiment of the present invention, user **interface** representations are stored in the user **interface** repository 152 and retrieved as needed from this repository by the business process layer 120. This is an appropriate mode of storing a large amount of user **interface** representations on a back-end database host, thus alleviating performance and space constraint problems on the client or business process hosts. However, for smaller applications, user **interface** representations may not need to be stored on a separate user **interface** repository 152. Accordingly, a user **interface** converter 212 can generate a header file 247 instead of database files 246...
- ...GUI files 248 are used by application developers and maintenance personnel to modify application screens and...
- ...of the re-engineering system 30. In a preferred embodiment of the present invention, the GUI file 248 are written in Microsoft Visual Basic. The application re-engineering process 30 then uses the GUI file to load screen information in Visual Basic, which can be viewed as the graphical user **interface** editor 310, make any modification in Visual Basic, resulting in a modified GUI file, and then run a Visual Basic to Oracle conversion process as described regarding the graphical user **interface** editor 310 to load the modified GUI file into the user **interface** repository database 152, ready for usage by the application...
- ...The user **interface** conversion utility 210 calls the user **interface** converter 212 to generate the "target" representation just described. In a preferred embodiment of the present invention, the user **interface** converter 212 is an ANSI C program, which takes a MFS file as an input and generates output files. To perform this function, the user **interface** converter 212 can be structured using conventional compiler technology, including a scanner 241, a parser...
- ...the statements of the source language. In this context, the delimiter that enables the user **interface** converter 212 to determine when the end of a statement has been reached is defined...
- ...calling process, eliminating the need to maintain a two-way communication structure with the user **interface** module and the accompanying state information in the business layer. Instead, re-architected batch processes are stand-alone programs constituted by a **wrapper** that provides means to parse the input arguments and call the top-level batch job...
- ...this top-level batch job requires some form of job scheduling infrastructure. As an example, **legacy** Job Control Language (JCL) can be converted to a scripting language-equivalent such as UNIX...

- ...commands or library calls that provide functionality that is similar to that of the source **legacy** system. In spite of these differences, batch conversion and processing follows the same fundamental principles ...
- ...target DDL language for increased maintainability and flexibility, as was the case with the user **interface** and procedural language conversion utility. For illustrative purposes, IMS DL/1 can be considered as...
- ...a high-level therefore, the custom and re-engineering system 30 includes a graphical user **interface** editor 310, a graphical business process editor 320, a graphical business object editor 330, a...
- ...the application code to be generated automatically from graphical representations. In particular, the graphical user **interface** editor 310 can be a commercially available user **interface** display platforms or GUI builders discussed in the context of the presentation layer 110...
- ...FIG. 28 is a block diagram of the graphical user **interface** editor 310 of FIG. 1. The graphical user **interface** editor 310 is a typical user **interface** made to create menus and paint screens. As such, the graphical user **interface** editor 310 includes a screen editor 311 to position graphical representations of business objects on ...
- ...labels, buttons, selection boxes, pull down lists, and similar graphical objects that compose a user **interface**. These graphical representations of business objects can be grouped so that a screen can be...
- ...be displayed across a number of application screens. The screen editor 311 creates internal user **interface** representations 312 which are processed by a user **interface** code generator 313 into data stored in the user **interface** repository 152...stored in the relational tables of the data layer RDBMS. As such, it is an **interface** that provides graphical access to each application table and permits the application developer or maintainer...

1/3,K/14 (Item 13 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4104970 **IMAGE Available

Derwent Accession: 1998-312857

Utility

E/ **System, method and article of manufacture for communications utilizing calling, plans in a hybrid network**

Inventor: Elliott, Isaac K., Colorado Springs, CO

Krishnaswamy, Sridhar, Cedar Rapids, IA

Assignee: MCI Communications Corporations 02), Washington, DC

MCI Communications Corp (Code: 40955)

Examiner: Chin, Wellington (Art Unit: 273)

Assistant Examiner: Carman, Melissa Kay

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 5867495 | A | 19990202 | US 96758734 | 19961118 |

Fulltext Word Count: 98158

Description of the Invention:

...physical network model provides QOS provisions such as wide range of qualities, adequate QOS for **legacy** applications, congestion management and user-selectable QOS...

...requirement in Service Creation is for the engineers who are producing basic capability objects to **insure** each can be reused in many different services as needed...The dbMon typically presents a passive **interface**; data is fed to it. However monitoring is a hierarchical activity and further analysis and...

...dbMon 2240 supports the following **interface** operations...

...The Operations consoles (Ops) 2244 provide the workstation-**interface** for the personnel monitoring, administering, and otherwise managing the system. The Ops consoles provide access...

1/3,K/15 (Item 14 from file: 654) Links
US PAT.FULL.
(c) Format only 2007 Dialog. All rights reserved.

4104969 **IMAGE Available
Derwent Accession: 1998-312857
Utility

E/ **System, method and article of manufacture with integrated
video conferencing billing in a communication system architecture**

Inventor: Krishnaswamy, Sridhar, Cedar Rapids, IA
Elliott, Isaac K., Colorado Springs, CO
Reynolds, Tim E., Iowa City, IA
Forgy, Glen A., Iowa City, IA
Solbrig, Erin M., Cedar Rapids, IA
Assignee: MCI Communication Corporation 02), Washington, DC
MCI Communications Corp (Code: 40955)
Examiner: Chin, Wellington (Art Unit: 273)
Assistant Examiner: Carman, Melissa Kay

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| | ----- | -- | ----- | ----- | ----- |
| Main Patent | US 5867494 | A | 19990202 | US 96752271 | 19961118 |

Fulltext Word Count: 125798

Description of the Invention:

...physical network model provides QOS provisions such as wide range of qualities, adequate QOS for **legacy** applications, congestion management and userselectable QOS...

...requirement in Service Creation is for the engineers who are producing basic capability objects to **insure** each can be reused in many different services as needed...The dbMon typically presents a passive **interface**; data is fed to it. However monitoring is a hierarchical activity and further analysis and...

...dbMon 2240 supports the following **interface** operations...

...The Operations consoles (Ops) 2244 provide the workstation-**interface** for the personnel monitoring, administering, and otherwise managing the system. The Ops consoles provide access....

1/3,K/16 (Item 15 from file: 654) Links

US PAT.FULL.

(c) Format only 2007 Dialog. All rights reserved.

4061950 **IMAGE Available

Derwent Accession: 1998-594401

Utility

EXPIRED

E/ Hybrid processor and method for executing incrementally
upgraded software

; COMPUTER APPARATUS

Inventor: Kirsch, Steven A., Agoura, CA

Mellema, Dwight J., Pasadena, CA

Assignee: Raytheon Company 02), Lexington, MA

Raytheon Co (Code: 69864)

Examiner: Donaghue, Larry D. (Art Unit: 273)

Combined Principal Attorneys: Alkov, Leonard A.; Lenzen, Jr., Glenn H.

| | Publication Number | Kind | Date | Application Number | Filing Date |
|-------------|-----------------------|------|----------|-----------------------|----------------|
| Main Patent | US 5828897 | A | 19981027 | US 96769571 | 19961219 |

Fulltext Word Count: 4699

Description of the Invention:

...shown as a functional block diagram that contains a DoD-STD 1750A processor 11 (or **legacy** processor 11) containing dual central processing units (CPUs) 12a, 12b and dual program memories 13a...

...local bus 18, allowing tightly-coupled functions to operate on the processors 11, 20. An **interface** 15 is provided as part of the DoD-STD 1750A processor 11 that couples the...

...input/output path for the processor 20 via its local bus 18 and an optional **interface** 19...

...in effect, creates a multi-ported memory 14 that is implemented using a relatively inexpensive **interface** circuit 16, such as a field programmable gate array 16, that interfaces the advanced processor...

...processor 11 are not accessing the memory 14, microtiming changes in the execution environment of **legacy** software 23 (FIG. 2) running on the central processing units 12a, 12b of the 1750A...

...processors 11, 20 provides a physical mechanism in which shared variables 24 are accessible to **legacy** and new application software 23, 25 (FIG. 2) executing on both processors 11, 20. However...

- ...by the software tools which compile and link the application software 23 running on the **legacy** processor 11, but then the application software 25 running on the 32-bit processor 20...
- ...the operand memory 14 using a native representation of the instruction set architecture of the **legacy** processor 11. However, these shared variables or operands 24 are typically incompatible with the instruction ...
- ...The **legacy** shared operand 24 implementation is thus hidden from the upgraded or new application software 25. Only knowledge of the **interface** specification of the shared operand 24 is required. As **legacy** software 23 is migrated to the new language, object (operand 24) implementations change, but do not affect the previously migrated code as long as the **interface** specification is maintained ...
- ...The present software architecture 30, in effect, creates a **wrapper** around the existing **legacy** software 23. The object request broker 21 manages the **wrapper** to **insure** a consistent object **interface** to the shared memory 14...
- ...the addition of a symmetrical object request broker (not shown) that is used by the **legacy** software 23. Shared objects (variables or operands 24) may then be stored in the operand...
- ...object request broker 21 is disposed 43 on the second processor 20 to provide an **interface** between the software 25 that runs on the second processor 20 and the reference depository...
- ...request broker (not shown) is disposed 47 on the first processor 11 to provide an **interface** between the software 23 that runs on the first processor 11 and the second reference...

1/3,K/17 (Item 1 from file: 761) Links
Datamonitor Market Res.
(c) 2007 Datamonitor. All rights reserved.
00037547

Global Round-up: 1.32 INDUSTRY COMMENT

Main Title: MarketWatch

Publication Date: December 06, 2005

Source: DATAMONITOR

Telephone: +44 20 7675 7000

Word Count: 4519 (2 pp.)

Language: English

Country: WORLD

Industry: MARKETING SERVICES

Company Names (DIALOG Generated): Actona ; Akimbi Systems ; Another Microsoft ; Apple ; Autonomy 's Intelligent Data ; Autonomy Corporation ; Autonomy Inc ; AutoZone ; Azul Systems ; ACNS ; AT&T ; Bank of America ; Blum Capital Partners ; Borland ; Both IBM ; Brasil Telecom ; Butler Group ; BT Global Services ; BT Infonet ; Cambridge Technology Partners ; Cardiff Software ; Celerant Consulting ; Cesky Telecom ; Chilean ; Cisco Systems ; Clearing Corporation ; Comicro ; Community Technology ; Compute Appliances ; Computer Associates ; Computer Reseller News ; Content Networking Software ; Credit Suisse First Boston ; Dralasoft ; Eircom Group ; Enterprise Resource Planning ; Equally Cisco ; European Commission ; European Software Association ; Expand Networks ; EMC ; EMEA ; FolderShare ; Follow Novell ; Further M & A ; FTC ; General Motors ; Global Address List ; Golden Gate Capital ; Goodyear ; Google ; Grupo Santander ; Hallmark Cards ; Hewlett Packard ; Infor Global ; Informix ; Internet Corp ; IBM Global Services ; ICANN ; ING ; ISP ; IT Management ; Java Virtual Machine ; Legadero Software ; Lifetime Products ; Lilly Software ; Live Safety Center ; Lumin ; Luminating ; Macnica Networks ; Mercury Interactive ; Messaging Management ; Meteor Mobile Communications ; Microsoft 's Media Center ; Microsoft Business Solutions ; Microsoft Corp ; Microsoft Exchange ; Microsoft Office ; Motion ; Motorola ; Nokia ; Novell ; NCR Corporation ; Open Invention Network ; Open Source Development Labs ; Oracle ; Overstock ; Pearl Group ; Philips ; Pixar ; Providence Equity Partners ; RealNetworks ; Red Hat ; Royal Philips Electronics ; Security Center ; Sony ; Sun 's Open Office ; Sun Microsystems ; Sunopsis ; Surgient ; Symantec Corp ; SCO ; Tally Systems Corp ; Tata Consultancy Services ; Teradata ; Time Warner ; Trend Micro ; TCS/Pearl Group ; TFT ; TPI ; Unisys ; US Federal Trade Commission ; Veritas ; Verity ; VeriSign ; Virtual Machine ; Vodafone ; Webroot Software ; Windows Live Safety Center ; WAFS ; WAN ; WDS ; Yahoo

...the operation of life and pensions administration for the closed book portfolio of UK-based insurer Pearl Group. The deal involves the transfer of existing Pearl non-IT staff to TCS...

...950 of Pearl Group's 1,100 staff, as well as leasing buildings at the **insurer's** existing site in Peterborough, UK. TCS plans to establish a business process outsourcing (BPO...

...an overhead in terms of the cost of extra systems and business administration required. Another **insurer**, Swiss Re, set up a subsidiary around eight years ago to buy such policies, and...into one single solution, enabling users to gain connectivity through an easy-to-use client **interface** and, depending on location, to exploit a number of different access technologies, such as Wi...

...Vista CTP that anti-spyware would be bundled is the fact that the Security Center **interface**, introduced in Windows XP Service Pack 2, now contains a reference to spyware alongside anti...as cutting costs by targeting Novell's two corporate jets, its "overstaffed" R&D department, **legacy** products, and its 400 or so NetWare engineers, as well as selling non-core businesses...

...in October.

The question is, where next for cost cutting at Novell? As well as **legacy** sales positions, clearly the R&D department should be a focus, according to Blum Capital...

...web standards that enables features like drag-and-drop components. AJAX enables a slick user **interface** that works more like a desktop app than a typical web site. There are no...

...respects, all of the aforementioned Live services, including Office Live (see separate article), are a **wrapper** around a core search-advertising engine, Live Search and adCenter.

MSN, currently Microsoft's flagship...business and allow for flexible growth opportunities for the mobile unit with no shackles to **legacy** communications. It sold its Eircell operation to mobile giant Vodafone for E4.3 billion in...

1/3,K/18 (Item 1 from file: 991) Links

NewsRoom 2006

(c) 2007 Dialog. All rights reserved.

1189559964 178C1UKV

From promises to prime time: early adopters got into the Web-services game a while ago, but most insurers took a wait-and-see stance and held off for proof of performance. The time may now be ripe to get on board--but jumping on without a good strategy could make for a rough ride.(TECHNOLOGY)

Meyer, David

Risk & Insurance , v 17 , n 5 , p 74

Saturday , April 15, 2006

Journal Code: ARBR **Language:** English **Record Type:** Fulltext

Document Type: Magazine **ISSN:** 1050-9232

Word Count: 1,501

...recently, the majority of Web services have been implemented in their simplest form with some **insurers** using services as a **wrapper** for their **legacy** system to expose the older technology to the Internet--a technique commonly referred to as Web-enabling. While the resulting user **interface** may be based on Web services, the underlying technology remains the **legacy** architecture with all its maintenance and integration challenges.

Web services have also been commonly used...

